

32.973.73

Т 48

В. М. ТКАЧУК, О. М. ТКАЧУК

ПРАКТИКУМ НА ЕОМ

Навчальний посібник

ЧАСТИНА 1

ВИДАВНИЧА СИСТЕМА *LaTeX*

МІНІСТЕРСТВО ОСВІТИ І НАУКИ,
МОЛОДІ ТА СПОРТУ УКРАЇНИ
ПРИКАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАСИЛЯ СТЕФАНИКА

В. М. ТКАЧУК, О. М. ТКАЧУК

Навчальний посібник

ПРАКТИКУМ НА ЕОМ

ЧАСТИНА 1

ВИДАВНИЧА СИСТЕМА *LaTeX*

НБ ПНУС



785725

Івано-Франківськ
Прикарпатський національний університет імені Василя Стефаника
2012

Рекомендовано до друку вченою радою факультету математики та інформатики
Прикарпатського національного університету імені Василя Стефаника,
(протокол №2 від 16 жовтня 2012 р.)

Рецензенти:

Соломко А.В. – кандидат фізико-математичних наук, доцент кафедри математичного і функціонального аналізу факультету математики та інформатики Прикарпатського національного університету імені Василя Стефаника;

Махней О.В. – кандидат фізико-математичних наук, доцент кафедри диференціальних рівнянь і прикладної математики факультету математики та інформатики Прикарпатського національного університету імені Василя Стефаника.

Ткачук В. М.

Т48 Практикум на ЕОМ, Частина 1 Видавнича система *LaTeX* / В. М. Ткачук., О. М. Ткачук. – Івано-Франківськ : Видавництво Прикарпатського національного університету імені Василя Стефаника, 2012. – 178 с.

В посібнику розглянуто систему комп'ютерного набору *LaTeX*, яка широко використовується для створення високоякісних текстових документів фізико-математичного спрямування. Пакет *LaTeX* є вільнорозповсюджуваним, поширюється та регулярно оновлюється через мережу Internet.

Посібник складається із лабораторних робіт, тематика яких відповідає робочій програмі базового курсу «Практикум на ЕОМ» для студентів 2-го курсу спеціальності «Математика». Кожна робота розрахована на самостійне виконання студентом. Інформація та приклади, що ілюструють теоретичний матеріал, надані оригінальними студентами, що мають мінімальні навички роботи і може використовуватися при проведенні лабораторних та практичних занять для студентів фізико-математичних напрямків підготовки.

НАУКОВА БІБЛІОТЕКА
код 0215288
ІНБ. № 78 57 25

УДК 681.3.06
ББК 32.973-018.1 72.4 74 76.17

© Ткачук В. М., Ткачук В. М., 2012
© Видавництво Прикарпатського національного університету імені Василя Стефаника, 2012

Лабораторна робота №1

Тема роботи: Видавнича система LaTeX

Мета роботи: формування навичок роботи по створенню та компіляції документів видавничої системи *LaTeX*. Знайомство із інтегрованим середовищем *WinEdt (Kile)*

Для виконання роботи необхідно знати:

- призначення видавничої системи *LaTeX*;
- команди компіляції та перетворення форматів документу;
- інтерфейс середовища *WinEdt (Kile)*;
- прийоми введення та редагування документу;
- загальну структуру вихідного файлу;
- компіляцію документу: порядок проведення та виправлення помилок в командному рядку;
- основні формати файлів, що використовуються в *LaTeX* та їх призначення;
- вставку коментарів у документ *LaTeX*;
- типові помилки, що можуть виникати при компіляції документу.

Історична довідка

Перша версія системи комп'ютерної верстки *TeX* була випущена в 1979 році американським професором математики Дональдом Кнотом (*Donald Ervin Knuth*). Вона призначена для верстки тексту та математичних формул на високому типографському рівні. В *TeX* користувач задає тільки текст та його структуру, система сама на базі вибраного шаблону формує кінцевий документ, виконуючи функції і дизайнера, і верстальника.

У його сучасному вигляді *TeX* вийшов у світ в 1982 році. Деякі вдосконалення були зроблені у 1989 році для підтримки 8-бітних символів і багатьох мов. *TeX* відомий стабільністю, здатністю працювати на багатьох комп'ютерних платформах і операційних системах, відсутністю помилок.

LaTeX (<http://www.latex-project.org/>) (читається як «латех») – надбудова над системою *TeX* у вигляді набору макропакетів та макросів є системою більш високого рівня, ніж *TeX*. *LaTeX* був написаний Леслі Лампортом в 1984 році (*Leslie Lamport*).

У 1994 році пакет *LaTeX* був оновлений командою *LaTeX3* на чолі з Франком Міттельбахом (*Frank Mittelbach*), з метою вдосконалення версії *LaTeX* 2.09. Щоб не плутати цю версію зі старою, вона отримала назву

LaTeX2ε. З 1998 року діє офіційний стандарт з підтримкою кирилиці в TeX/LaTeX (включно з українською мовою).

Теоретичні відомості

Система комп'ютерного набору LaTeX – це вільнорозповсюджуване програмне забезпечення, призначене для підготовки та створення науково-технічної документації високої типографської якості. Вона не є системою типу WYSIWYG (What You See Is What You Get: що друкую – те і бачу) і для перегляду, як буде виглядати документ, необхідно запустити окрему програму та скомпілювати документ. Вихідним документом для LaTeX є звичайний текстовий ASCII файл (текстовий файл із розширенням *.tex*, підготовлений за допомогою будь-якого текстового редактора) із текстом документу та командами його форматування.

Для отримання кінцевого документу (готового до перегляду чи виводу на друк) вихідний файл необхідно скомпілювати за допомогою програми-транслятора: в результаті буде створено файл із розширенням *.dvi*, який можна переглянути чи видрукувати за допомогою спеціальної програми, або перетворити до одного із загальноприйнятих форматів (див таб. №2). Для вводу команд компіляції найзручніше за допомогою *Блокноту* створити *bat* – файл (файл із розширенням *.bat*), який розмістити у папці із вихідним документом. Наприклад, для компіляції вихідного файлу *file.tex* необхідно створити та виконати файл із розширенням *.bat*, як це показано на рис.1. Ім'я файлу принципового значення не має.

Перелік команд для компіляції та перетворення форматів документів приведено в таб. №1.

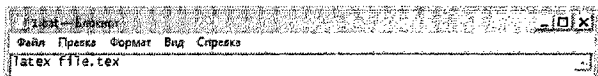


Рис.1. Приклад створення файлу *l.bat* для компіляції документу *file.tex* та отримання готового до перегляду файлу *file.dvi*.

Таблиця №1. Команди для роботи із вихідним документом LaTeX

Команда	Короткий опис
<i>latex file.tex</i>	Компіляція <i>file.tex</i> та отримання файлу формату <i>.dvi</i> із іменем вихідного документу.
<i>dvips file.dvi</i>	Перетворення файлу <i>file.dvi</i> в файл формату <i>.ps</i> .

<i>dvipdfm file.dvi</i>	Перетворення файлу <i>file.dvi</i> в файл формату <i>.pdf</i> .
<i>yap file.dvi</i>	Перегляд файлу <i>file.dvi</i> за допомогою стандартного переглядача <i>Yap</i>
<i>pdflatex file.tex</i>	Компіляція вихідного документу <i>file.tex</i> та створення файлу формату <i>.pdf</i> . Документ можна переглянути, наприклад, за допомогою <i>Acrobat Reader</i> .

Основні формати файлів, що використовуються при роботі із видавничою системою, зібрано в таб. №2.

Таблиця №2. Основні формати файлів та їх призначення

Формат файлу	Короткий опис
<i>.tex</i>	Вихідний LaTeX документ
<i>.dvi</i>	<i>DeVice Independent</i> (апаратно незалежний) — формат файлу видавничої системи LaTeX. Створюється на етапі компіляції для перегляду скомпільованого документу.
<i>.log</i>	Файл, у який виводиться інформація про хід компіляції документу (у текстовому форматі).
<i>.toc</i>	Містить назви розділів та інших структурних елементів. Використовується для побудови змісту документу.
<i>.lof</i>	Служить для побудови списку ілюстрацій, що використовуються в документі.
<i>.lot</i>	Служить для побудови списку таблиць, що створені в документі.
<i>.aux</i>	Файл передає інформацію від одного проходу до наступного: для отримання кінцевого документу може знадобитися кілька процесів компіляції документу. Використовується для збереження інформації про перехресні посилання в межах цілого документу, який може складатися із кількох окремих файлів.
<i>.pdf</i>	<i>Portable Document Format</i> (відкритий формат файлу) – представлення документів у незалежному від пристрою виведення та роздільної здатності вигляді. Містити повну інформацію про документ, таку як: текст, зображення, векторні зображення, відео, інтерактивні форми та ін.

.ps (.eps)	<i>Encapsulated Post Script</i>) —графічний файл у машинних кодах на мові програмування PostScript. Використовується переважно для друку. Може містити як векторну, так і растрову інформацію.
-------------------	---

Для внесення будь-яких змін у документ необхідно мати вихідний **LaTeX** документ (документ із розширенням *.tex*).

Інформація про хід компіляції відображається у окремому вікні (див рис.2.) та виводиться у файл із розширенням *.log* та іменем вихідного документу (протокол трансляції знаходиться у папці, де і сам документ). Якщо вихідний файл створено із порушеннями синтаксису **LaTeX**, то процес компіляції буде призупинено на першій-же помилці.

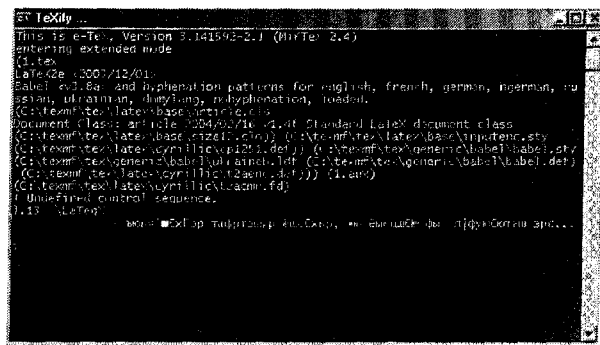


Рис.2. Вигляд вікна, в якому відображається хід компіляції документу

Помилки можуть бути виправлені як у ході компіляції, так і у вихідному документі. Оскільки перша помилка може призводити до появи наступних, то процес правки повинен носити поступовий характер, починаючи від першої. Помилки на етапі компіляції можна і проігнорувати, але при цьому, як правило, документ на екрані буде відображено невірно. Перелік команд для управління ходом компіляції документу зібрано в таб. №3.

Таблиця №3. Основні команди етапу компіляції документу

Команда	Короткий опис
H + Enter	Вивід додаткової інформації про помилку

X + Enter	Негайне припинення процесу компіляції документу
I + Enter	Виправлення помилки у вікні компіляції
R + Enter	Ігнорування всіх помилок, що виникають в ході компіляції документу

По завершенню трансляції документу (якщо помилок не виявлено) вікно буде закрито. При виявленні помилок трансляція документу призупиняється та виводиться наступна інформація:

- перший рядок повідомлення починається знаком оклику, після якого слідує коротка вказівка на характер помилки;
- другий рядок повідомлення починається з *I.*, після якого йде номер рядка документу, в якому знайдено помилку. Після номера міститься або весь текст рядка, або його частина до моменту появи помилки. *При виводі кирилиці текст документу буде відображатися некоректно;*
- третій рядок починається знаком питання, після якого необхідно задати одну із команд таб. 3. При натисненні **Enter** (без команди) компілятор сам спробує виправити знайдену помилку на свій розсуд та продовжить процес компіляції до наступної помилки.

Огляд типових помилок, що виникають при компіляції документу, приведено в Додатку №1.

Більш зручним для формування вихідного документу є спеціалізовані редактори, які являють собою інтегроване середовище розробки та використовуються для створення документу **LaTeX**, перевірки його орфографії, компіляції і виправлення помилок та перегляду отриманого результату у одному із загальноприйнятих форматів. Вони дозволяють вводити більшість команд форматування тексту за допомогою пунктів меню чи відповідних кнопок робочого столу програми. Одним із найпопулярніших для роботи в **OC Linux** є вільнодоступний редактор **Kile** (див. Додаток №2). Для роботи в **OC Windows** підійде, наприклад, **WinEdt** (див. Додаток №3), використання якого вимагає ліцензії.

Загальна структура документу

Вхідний документ **LaTeX** завжди починається преамбулою, яка, в свою чергу, починається заголовком:

\documentclass[options]{class}

class - клас документу: визначає загальну структуру та особливості форматування документу;

options - необов'язковий аргумент: дозволяє міняти значення ряду параметрів та деяких правил форматування, які приймаються по замовчуванню для даного класу.

Стандартними класами **LaTeX** для створення документів є:

- article**: стаття;
- book**: книга;
- report**: доповідь;
- letter**: лист;
- slides**: слайд.

Основні класи документів більш детально будуть розглянуті у наступних лабораторних роботах. Приклад заголовка, який задає стиль оформлення статті із розміром шрифту **12pt** (необов'язковий параметр):

```
\documentclass[12pt]{article}
```

Тут і надалі у вихідному документі при введенні команд форматування великі та малі букви мають значення.

В **LaTeX** можуть використовуватися будь-які одиниці довжини, приведені у таблиці нижче:

Одиниця довжини	Значення
cm	сантиметр
mm	міліметр
in	дюйм (2.54 сантиметра)
pt	пункт (0.35 міліметра)
pc	піка (12 пунктів)

Всі вони є рівноправними, але навіть при їх нульовому значенні обов'язково вказати одиниці вимірювання.

Після заголовку іде власне преамбула документу, де задаються параметри та команди, що будуть стосуватися всього документу. Для розширення базової версії **LaTeX** (завантаження додаткових файлів, які містять визначення нових команд чи переозначають старі) використовується декларація **\usepackage**:

```
\usepackage[options]{package}
```

package – ім'я пакету, що підключається.

Сам текст документу вводиться після преамбули між командами **\begin{document}** та **\end{document}**. Все, що знаходиться після команди **\end{document}** на етапі компіляції буде проігноровано.

Основні правила вводу тексту:

- текст вводиться без переносів слів (**LaTeX** розставить їх сам);
- слова розділяються пробілами, а їх кількість значення не має (всі лишні пробіли при компіляції будуть проігноровані);
- кінець рядка сприймається як пробіл;
- абзаци розділяються між собою порожніми рядками, а їх кількість значення не має.

Для роботи із кириличним текстом у преамбулі документу необхідно підключити додаткові пакети (залежить від операційної системи, в якій ви працюєте):

Windows (для відображення кирилиці використовується сторінка-кодування **1251**):

```
\usepackage[cp1251]{inputenc}
```

Linux (для відображення кирилиці використовується **utf8** кодування):

```
\usepackage[utf8]{inputenc}
```

Для встановлення кодування текстових шрифтів **T2A** із українськими буквами, переозначення стандартних стильових заголовків, встановлення правил переносу українських слів, введення нових команд для набору символів, специфічних для української мови необхідно підключити пакет **babel** із опцією **ukrainian**:

```
\usepackage[ukrainian]{babel}
```

Якщо в документі використовується декілька мов, то через кому задаємо словники, які необхідно використовувати. Наприклад:

```
\usepackage[english,russian,ukrainian]{babel}
```

Визначальною при компіляції документу буде мова, що стоїть останньою у переліку. У наведеному вище прикладі це українська. Якщо визначальною мовою документу є англійська, то додатково необхідно підключити пакет, який дасть можливість використовувати в ньому кирилицю:

```
\usepackage[T2A]{fontenc}
```

Тут і надалі приклади документів будуть сформовані для роботи в ОС **Windows**. Для використання наведених прикладів при роботі в **Linux**

необхідно замінити при підключення пакету кодову сторінку **cp1251** на **utf8**.

Приклад вихідного LaTeX - документу із оформленням у стилі «стаття» та розміром шрифту 12pt:

```
\documentclass[12pt]{article}
\usepackage[cp1251]{inputenc}
% \usepackage[utf8]{inputenc} для ОС Linux
\usepackage[T2A]{fontenc}
\usepackage[english,ukrainian]{babel}
\title{Це є перший мій перший документ}
\author{Автор статті - студент...}
\begin{document}
\maketitle

\begin{abstract}
Анотація до документу, що ілюструє найпростіші
можливості LaTeX
\end{abstract}
```

LaTeX - комп'ютерна видавнича система, що служить для підготовки наукової документації. Вихідний документ для LaTeX являє собою файл (текстовий файл із розширенням .tex, підготовлений за допомогою будь-якого текстового редактора) із текстом документу та командами його форматування.

Для отримання кінцевого документу вихідний файл необхідно скомпілювати за допомогою програми-транслятора: в результаті буде створено файл із розширенням .dvi,

який можна переглянути за допомогою спеціальної програми.

Для внесення будь-яких змін у документ обов'язково необхідно мати вихідний LaTeX документ.

```
\end{document}
```

Результат компіляції документу:

Це є перший мій перший документ

Автор статті - студент...

12 січня 2012 р.

Анотація

Анотація до документу, що ілюструє найпростіші можливості LaTeX

LaTeX - комп'ютерна видавнича система, що служить для підготовки наукової документації. Вихідний документ для LaTeX являє собою файл (текстовий файл із розширенням .tex, підготовлений за допомогою будь-якого текстового редактора) із текстом документу та командами його форматування.

Для отримання кінцевого документу вихідний файл необхідно скомпілювати за допомогою програми-транслятора: в результаті буде створено файл із розширенням .dvi, який можна переглянути за допомогою спеціальної програми. Для внесення будь-яких змін у документ обов'язково необхідно мати вихідний LaTeX документ.

Для введення команд форматування тексту в LaTeX зарезервовано ряд спеціальних символів, які просто-так в документ вставлені бути не можуть:

Спеціальний символ	Призначення
{ та }	Створення групи у вихідному документі.
\$	Використовується при введенні математичних формул.
%	Вставка коментарів у вихідний документ. На етапі компіляції весь рядок після даного символу ігнорується.
_ та ^	Нижній та верхній індекс при вводі формул.
~	Нерозривний пробіл – заборона розриву при перенесенні слова на наступний рядок.
\	Початок вводу команд форматування.

Примітка: для вставки даних символів у текст документу перед ним необхідно набрати символ `\`. Наприклад, для вводу у текст документу знаку долара необхідно набрати `\$`. Для вставки ряду символів необхідно скористатися спеціальними командами:

- символ `^`: команда `^{};`
- символ `~`: команда `~\sim&;`
- символ `\`: команда `\backslash&;`

Ввід команд форматування

Для форматування тексту документу використовують команди, які починаються символом `\`. Команди можуть бути двох типів:

- після `\` слідує символ, що не є літерою;
- після `\` слідує ім'я команди.

Загальні правила набору команд:

- після `\` пропуск є недопустимим;
- команда розміщується тільки в одному рядку;
- великі та малі букви розрізняються;
- після імені команди обов'язково слідує пропуск.

Обов'язкові аргументи задають у фігурних дужках та опущені бути не можуть: при їх відсутності на етапі компіляції буде видана помилка. У квадратних дужках, через кому, можуть бути задані необов'язкові аргументи. Наприклад (створення документу у стилі «стаття»; необов'язкові параметри задають розмір шрифту **12pt** та оформлення тексту у дві колонки):

```
documentclass[12pt, twocolumn]{article}
```

Параметри

Задання параметрів починається символом `\` та використовується для зміни значень величин, якщо запропоновані **LaTeX** стильові значення не підходять. Наприклад, для встановлення величини абзацного відступу **1.5** сантиметрів необхідно в преамбулі додати наступний рядок:

```
\parindent = 1.5cm
```

Створення групи

Для застосування певних параметрів та команд до фрагменту тексту використовуються фігурні дужки, які утворюють групу. Фігурні дужки повинні бути збалансованими: кількість відкриваючих дужок рівна кіль-

кості закриваючих, бо інакше буде видано повідомлення про помилку. Команди форматування, які використовуються в середині групи, діють тільки в її межах та не впливають на решту тексту. Допускається вкладення одних груп у інші (але не їх перехресне вкладення). Деякі із команд форматування можуть бути глобальними та зберігати свою дію і за межами групи. Приклад створення групи:

Приклад створення групи.

```
{\bf Даний текст буде надруковано жирним шрифтом}.
```

Даний текст знаходиться за межами групи, тому буде надруковано звичайним шрифтом.

```
{\esh Приклад створення {\bf другої} групи виділеним шрифтом}
```

Результат компіляції фрагменту:

Приклад створення групи. Даний текст буде надруковано жирним шрифтом. Даний текст знаходиться за межами групи, тому буде надруковано звичайним шрифтом.

Приклад створення другої групи виділеним шрифтом

Створення оточення

Оточення – це фрагмент документу, що знаходиться між командами `\begin{Ім'я оточення}` та `\end{Ім'я оточення}`. Ім'я оточення є обов'язковим параметром, а саме оточення - групою. В оточенні можна після обов'язкових задати і ряд необов'язкових параметрів. Наприклад використання оточення для розміщення тексту по центру:

```
\begin{center}
```

Приклад створення оточення для

вирівнювання тексту по центру.

```
\end{center}
```

Результат компіляції:

Приклад створення оточення для вирівнювання тексту по центру.

Коментарі до документу

Все, що слідує у рядку після % при компіляції буде проігноровано: це є однорядковий коментар. Пакет *verbatim* (підключається в преамбулі документу командою `\usepackage {verbatim}`) дозволяє використовувати оточення `\begin{comment} . . . \end{comment}` для створення багаторядкових коментарів. Наприклад:

Текст, що знаходиться в межах оточення comment на етапі компіляції буде опущений. Для роботи даного оточення у преамбулі документу обов'язково необхідно підключити пакет verbatim.

`\begin{comment}`

Приклад багаторядкового коментарю в документі. Даний фрагмент тексту на етапі компіляції буде опущений.

`\end{comment}`

Символ відсотка задає однорядковий коментар - текст у рядку, що іде після нього при компіляції документу буде проігнорований.

% Все після даного символу на етапі компіляції буде
% проігноровано

Результат компіляції:

Текст, що знаходиться в межах оточення comment на етапі компіляції буде опущений. Для роботи даного оточення у преамбулі документу обов'язково необхідно підключити пакет verbatim

Символ відсотка задає однорядковий коментар - текст у рядку, що іде після нього при компіляції документу буде проігнорований.

Хід роботи

1. Ознайомтесь зі основними принципами побудови документів в системі **LaTeX**. Вивчіть загальну структуру **LaTeX**-документу.
2. Введіть приведені приклади документів у **Блокноті**, скопіюйте їх, та перегляньте отримані результати, використовуючи команди таб. №1. При необхідності виправте помилки, що виникають при компіляції документу (див. Додаток №1).

3. Ознайомтесь із інтерфейсом середовища **WinEdt (Kile)** (див. Додаток №2 або №3), його можливостями, порядком створення і компіляції документу. Скопіюйте підготовлені раніше документи.

Контрольні питання

1. Які команди для роботи з документом **LaTeX** у командному рядку ви знаєте?
2. Яке призначення інтегрованого середовища **Kile (WinEdt)** ?
3. Як скопіювати та переглянути документ **LaTeX**?
4. Які основні формати файлів використовуються в **LaTeX**?
5. Що включає в себе загальна структура документу?
6. Що таке обов'язкові аргументи команди?
7. Для чого призначені необов'язкові параметри команди?
8. Які одиниці довжини можуть використовуватися при заданні значень параметрів?
9. Для чого використовуються спеціальні символи?
10. Як вставити спеціальний символ у текст документу?
11. Що таке оточення та його призначення?
12. Що таке група та її призначення?
13. Як вставити у документ однорядковий коментар?
14. Як вставити у документ багаторядковий коментар?

Лабораторна робота №2

Тема роботи: Основи верстки документів у видавничій системі LaTeX

Мета роботи: верстка документу у видавничій системі LaTeX з використанням стилів, параметрів сторінки та розділів документу

Для виконання роботи необхідно знати:

- основні стилі документу та їх призначення;
- налаштування параметрів сторінки;
- розділи документу та їх формування;
- способи задання нумерації сторінок;
- структуру документу у стилі статті;
- структуру документу у стилі листа.

Теоретичні відомості

Команда `\documentclass`, з якої починається будь-який LaTeX – файл, має один обов'язковий аргумент – назва основного стилю, який вказується у фігурних дужках після команди.

В LaTeX для оформлення документів розроблено наступні базові стилі (на їх основі розроблено значну кількість інших, похідних стилів):

- **article**: стаття – стиль оформлення наукових статей, звітів чи коротких документів. Він не має поділу документу на окремі глави. Титульна сторінка розміщується вверху першої сторінки. Стиль є найбільш прийнятним для формування більшості документів;
- **report**: доповідь – стиль для достатньо великих технічних документів (дипломних робіт, дисертацій і т.д.). Від попереднього відрізняється можливістю поділу документу на глави, а титульна сторінка формується на окремій сторінці;
- **book**: книга – стиль для видання книг. Документ формується виходячи з того, що в кінцевому варіанті він буде друкуватися на двох сторонах листа;
- **letter**: лист - стиль для написання ділових листів. Містить всі елементи для створення добре відформатованого листа: адреса; дата; підпис і т.д. В преамбулі перед текстом листа оголошують ім'я, адресу та інші атрибути відправника;
- **beamer**: створення електронних презентацій із можливістю використання стандартних стилів оформлення, застосування анімаційних ефектів та кнопок управління переглядом.

Кожен із перерахованих стилів має свої особливості, однак всі вони побудовані за одною схемою. Відмінності полягають тільки в деяких моментах форматування документу. Кожен із стилів має ряд стильових опцій (необов'язкових параметрів, значення яких приведено в таб. №4.). Їх значення можуть бути змінені в преамбулі документу.

Таблиця №4. Стильові опції та їх можливі значення

Параметр стилю	Можливі значення	Значення по замовчуванню
Стандартний розмір шрифту	<i>10pt, 11pt, 12pt</i>	<i>10pt</i>
Розмір паперу	<i>a4paper, a5paper, b5paper, letterpaper, legalpaper</i>	<i>a4paper</i>
Орієнтація сторінки	<i>landscape</i> – альбомна	Книжна
Наявність титульної сторінки	<i>titlepage</i> – так <i>notitlepage</i> – ні	<i>titlepage</i>
Положення номера у винесених формулах	<i>leqno</i> – номер зліва від формули	Справа
Одно- чи двосторонній друк	<i>oneside</i> – односторонній, <i>twoside</i> – двосторонній	<i>oneside</i> , але для класу <i>book</i> – <i>twoside</i>
Розміщення тексту в колонках	<i>onecolumn</i> – одна колонка, <i>twocolumn</i> – дві колонки	<i>onecolumn</i>
Сторінка, з якої може починатися нова глава	<i>openright</i> – тільки з правої сторінки розвороту, <i>openany</i> – із будь-якої	Для стилю <i>book</i> – <i>openright</i> , для інших – <i>openany</i>
Стиль виводу титульної сторінки	<i>titlepage</i>	Команда <i>maketitle</i> при наявності цього параметру друкує титульну сторінку на окремій сторінці

Опції стилів задаються перед назвою стилю в квадратних дужках. Якщо використовується кілька опцій, то вони розділяються комами. Наприклад:

```
\documentclass[10pt, a4paper, twocolumn]{article}
```

Стиль оформлення сторінок документу

Стиль оформлення сторінок документу задається командою `\pagestyle`, яка повинна знаходитися у преамбулі документу та може мати один із наступних обов'язкових аргументів:

- **empty** – номери сторінок не виводяться;
- **plain** – номери сторінок виводяться внизу по середині рядка;
- **headings** – номери сторінок виводяться вгорі сторінки (одночасно виводиться і назва відповідного розділу).

Команда `\thispagestyle` дозволяє поміняти стиль оформлення окремої сторінки. Вона аналогічна `\pagestyle`, але стосується тільки тієї сторінки, на яку попадає текст, що її оточує.

Формат відображення номерів сторінок можна поміняти за допомогою команди `\pagenumbering` із параметрами, значення яких приведено в таб. №5.

Таблиця №5. Можливі значення параметрів для задання стилю відображення номерів сторінок

Параметр	Стиль відображення
arabic	Арабські цифри (1,2,3,...)
roman	Римські малі цифри (i,ii,iii,...)
Roman	Римські великі цифри (I,II,III,...)
alph	Рядкові букви (a,b,c,...)
Alph	Заголовні букви (A,B,C,...)

Поля та розміри сторінок

В стандартних стилях значення більшості параметрів визначено наперед (див рис.3 та таб. №6). Якщо значення параметрів необхідно змінити, то в преамбулі документу задають відповідні параметри та їх нові значення. Обов'язковим є задання одиниць вимірювання (навіть якщо значення рівне нулю!).

Ширина тексту:

задається параметром `\textwidth`. При виводі тексту у дві колонки включає в себе ширину двох колонок та проміжок між ними. Наприклад, щоби ширина тексту на сторінці була рівна **310pt** необхідно в преамбулі документу задати (аналогічно можуть бути змінені всі інші параметри):

```
\textwidth{310pt}
```

Ширина лівого поля:

задається командою `\oddseademarking`. Поле відраховується не від краю сторінки, а від попереднього відступу 1 дюйм. При двосторонньому наборі тексту параметр задає ліве поле для сторінок із непарними номерами. Поля для сторінок із парними номерами можуть бути задані за допомогою команди `\evensidemargin`. При двосторонньому наборі `\columnsep` задає відстань між колонками.

Висота верхнього поля:

задається командою `\topmargin`. Поле відраховується не від краю сторінки, а від попереднього відступу 1 дюйм.

Висота тексту:

задається аналогічно до ширини тексту параметром `\textheight`.

Зсув сторінки в цілому:

зсув по горизонталі (при виводі на друк) задається параметром `\hoffset`;

зсув по вертикалі (при виводі на друк) задається параметром `\voffset`.

Допустимим є використання від'ємних значень параметрів – зсув відбудеться в протилежну сторону.

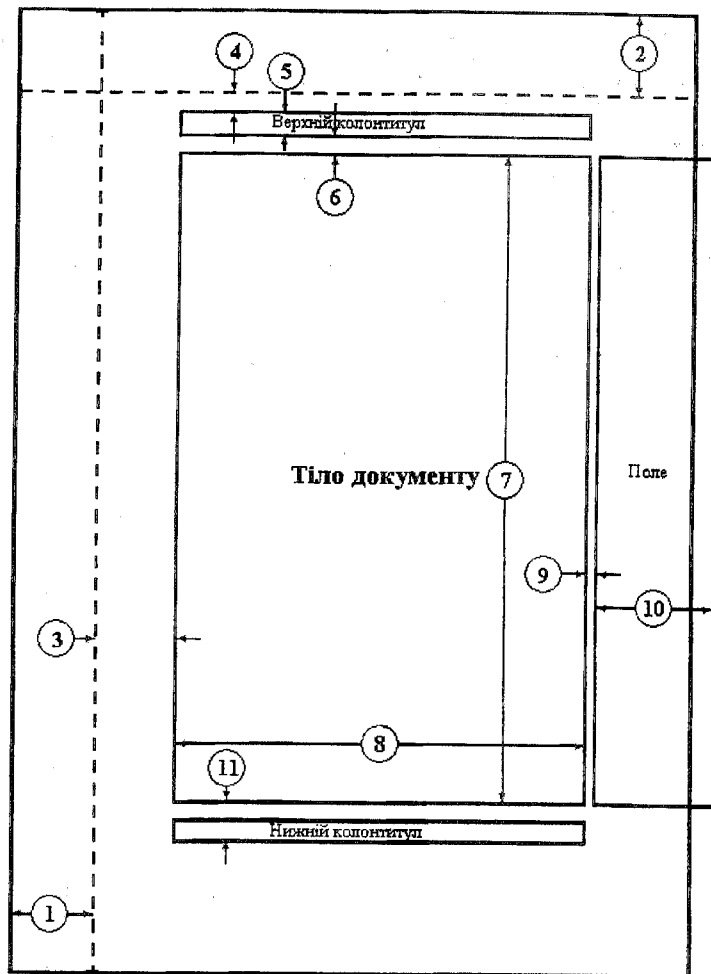


Рис.3. Параметри, що задають поля та розмір сторінок

В таб. №6 наведено параметри, що відповідають за розміщення тексту на сторінці та їх значення по замовчуванню. Нумерація параметрів в таблиці відповідає номерам позицій на рис. 3.

Таблиця №6. Параметри та їх значення по замовчуванню

Номер	Параметр	Значення по замовчуванню
1.	1 дюйм+ <code>\hoffset</code>	1 дюйм
2.	1 дюйм+ <code>\voffset</code>	1 дюйм
3.	<code>\oddseademarning</code>	72pt
4.	<code>\topmarning</code>	18pt
5.	<code>\headheight</code>	12pt
6.	<code>\headsep</code>	25pt
7.	<code>\textheigth</code>	568pt
8.	<code>\textwidth</code>	360pt
9.	<code>\marginparsep</code>	10pt
10.	<code>\marginparwidth</code> <code>\marginparpush=5pt</code> (не показано) <code>\voffset=0pt</code> <code>\paperheight=845pt</code>	103pt
11.	<code>\footskip</code> <code>\hooffset=0pt</code> <code>\paperwidth=597pt</code>	30pt

Заголовки та нумерація розділів

Для формування окремих розділів документу використовують наступні команди:

`\part` (частина);
`\chapter` (глава - не визначена для стилю `article`);
`\section` (розділ);
`\subsection` (підрозділ першого рівня);
`\subsubsection` (підрозділ другого рівня);
`\paragraph` (параграф);
`\subparagraph` (підпараграф).

Кожна наступна команда означає формування більш дрібного підрозділу, ніж попередній. Кожна команда має один обов'язковий параметр – назва підрозділу у фігурних дужках. Для початку нового розділу без його нумерації необхідно після імені команди вставити символ *. Нумерація здійснюється автоматично, а при початку нового розділу відповідного рівня починається спочатку.

Титульна сторінка, заголовок та анотація

Для оформлення заголовка необхідно:

- задати в преамбулі документу інформацію для заголовку (автор, назва, дата, і.т.д.);
- згенерувати заголовок командою `\maketitle`.

Автор задається командою `\author` із обов'язковим параметром – ім'ям автора. Декілька авторів повинні бути розділені командою `\and`.

Заголовок створюється за допомогою команди `\title`.

Дата формується за допомогою команди `\date`. При відсутності команди дата все-одно буде виведена (дата компіляції документу). Щоби дата не виводилася, необхідно ввести команду `\date{}` із порожнім аргументом у фігурних дужках.

Порядок слідування команд є довільним, але обов'язково до `\maketitle`. До будь-яких елементів титульного листа можна додати примітку за допомогою команди `\thanks`, (розміщується в середині відповідної команди формування титульної сторінки) із обов'язковим параметром – текстом примітки (абзаци у примітці формуються за допомогою команди `\endgraf`). Самі примітки друкуються внизу титульного листа.

При необхідності сформувати титульну сторінку самостійно (без використання стандартного стилю) можна скористатися оточенням `titlepage`. Текст між `\begin{titlepage}` та `\end{titlepage}` буде формувати титульну сторінку, формат якого можна задати самостійно. Вона виводиться в одну колонку без колонтитулів та номеру сторінки, а наступний текст починатиметься із наступної сторінки.

Анотація до документу (в класах `article`, `report`) може бути створена за допомогою оточення `abstract` за допомогою команд `begin{abstract}` та `end{abstract}`. Для тексту автоматично буде згенеровано заголовок «Анотація» (якщо підключено український стилевий пакет).

Нижче приведено приклад формування типової статті, який демонструє використання основних стильових параметрів та формування титульної сторінки:

```
\documentclass[10pt,twocolumn]{article} % оформлення
% документу у дві колонки
\usepackage[cp1251]{inputenc}
\usepackage[ukrainian]{babel}
\textheight=24cm % висота тексту
\textwidth=16cm % ширина тексту
\oddsidemargin=0pt % відступ від лівого краю
\topmargin=-1.5cm % відступ від правого краю
\pagestyle{headings} % вивід номерів сторінок
% вгорі сторінки

\title{Основи верстки документу у видавничій системі
\LaTeX }
\author{Студент групи М-31 Василів Б.Д.}
\thanks{Прикарпатський національний
університет імені В.Стефаника}
\date{2 лютого 2012} % вивід дати в титульному листі
%початок документу
\begin{document}
\thispagestyle{empty} % не виводити номер на текучій
% сторінці

\pagenumbering{Roman}
\maketitle % сформувати заголовок документу
\begin{abstract} % початок анотації до документу
Цей документ ілюструє основні можливості
видавничої системи \LaTeX по верстці документу. В
ньому проілюстровано
верстку документу із використанням стилю,
параметрів сторінки та розділів документу
\end{abstract} % кінець анотації
\part{Все про \LaTeX}
\section{Основи верстки документів} % перший розділ
\subsection{Стилі \LaTeX }
В \LaTeX для створення документів розроблено наступні
класи:
```

Article: стаття – стиль оформлення наукових статей, звітів чи коротких документів.

Стиль не містить поділу документу на окремі глави. Титульна сторінка (отримується командою `\maketitle`) розміщується вверху першої сторінки.

Клас є найбільш прийнятним для формування більшості документів;

Book: книга - стиль для видання книг. Документ формується виходячи з того, що в кінцевому варіанті він буде друкуватися на двох сторонах паперу;

Report: доповідь - стиль для достатньо великих технічних документів (дипломних робіт, дисертацій і т.д.). Від попереднього відрізняється можливістю поділу документу на глави, а титульна сторінка формується на окремому листі;

`\subsection*{Приклад формування підрозділу без номеру}`

Якщо після імені команди стоїть зірочка, то буде згенеровано підрозділ відповідного підрівня без використання нумерації

`\section{Робота з абзацами}`

Для початку нового абзацу у текст достатньо вставити порожній рядок

`\subsection{Перенесення слів}`

`\LaTeX` сам розставляє переноси у текстовому документі, використовуючи підключений пакет `~babel`

`\subsubsection{Спосіб перенесення слів}`

`rightthyphenmin` - параметр, що відповідає за мінімальну кількість літер, яка може бути перенесена на наступний рядок

`\subsubsection{Нерозривний пробіл}`

Символ `\~` використовується для того, щоби вказати `\LaTeX`, в якому місці абзацу не можна починати новий рядок

`%кінець документу`

`\end{document}`

Будь-який текст, що знаходиться після `\end{document}`, при компіляції буде проігнорований

Результат компіляції документу:

Основи верстки документу у видавничій системі \LaTeX

Студент групи М-31 Василів Б.Д. *

2 лютого 2012

Анотація

Цей документ ілюструє основні можливості видавничої системи \LaTeX у верстці документу. В ньому проілюстровано верстку документу із використанням стилів, параметрів сторінки та розділів документу

Частина I

Все про \LaTeX

1 Основи верстки документів

1.1 Стилі \LaTeX

В \LaTeX для створення документів розроблено наступні класи:

Article: стаття - стиль оформлення наукових статей, звітів чи коротких документів. Стиль не містить поділу документу на окремі глави. Титульна сторінка (отримується командою) розміщується вгорі першої сторінки. Клас є найближчим прибутим для формування більшості документів;

Book: книга - стиль для видання книг. Документ формується виходячи з того, що в кінцевому варіанті він буде друкуватися на двох сторонах паперу;

Report: доповідь - стиль для достатньо великих технічних документів (дипломних робіт, дисертацій і т.д.). Від попереднього відрізняється можливістю поділу документу на глави, а титульна сторінка формується на окремому листі;

Приклад формування підрозділу без номеру

Якщо після імені команди стоїть зірочка, то буде згенеровано підрозділ відповідного підрівня без використання нумерації

* Прикарпатський національний університет імені В.Стефаника

2 Робота з абзацами

Для початку нового абзацу у текст достатньо вставити порожній рядок

2.1 Перенесення слів

\LaTeX сам розставляє переноси у текстовому документі, використовуючи підключений пакет `babel`

2.1.1 Спосіб перенесення слів

`rightthyphenmin` - параметр, що відповідає за мінімальну кількість літер, яка може бути перенесена на наступний рядок

2.1.2 Нерозривний пробіл

Символ використовується для того, щоби вказати \LaTeX , в якому місці абзацу не можна починати новий рядок

Формування документу у стилі ділового листа.

Стиль **letter** має ряд принципових відмінностей від інших стандартних стилів: у ньому визначено ряд команд, які є недоступними в інших стилях і навпаки (див. таб. №7). Так, наприклад, в ньому немає можливості формувати текст у дві колонки, створювати титульну сторінку і т.д.

Таблиця №7. Деякі команди стилю **letter**, не доступні в інших стандартних стилях документу

Команда	Короткий опис
<code>\address</code>	Адреса відправника (лист приватний). Окремі рядки адреси розділяються комами. Якщо команда відсутня, то на першій сторінці листа (там за звичай на фірмовому бланку містяться всі атрибути відправника) залишається порожнє місце – лист офіційний.
<code>\signature</code>	Текст підпису відправника.
<code>\name</code>	Атрибути відправника (у зворотній адресі).
<code>\location</code>	Уточнює адресу відправника, якщо лист друкується на офіційному бланку організації. Виводиться внизу листа.
<code>\telephone</code>	Уточнює номер телефону відправника, який вказується на бланку, якщо лист є офіційним. Виводиться внизу листа
<code>\opening</code>	Власне текст листа (звернення до адресата). Завершується командою <code>\closing</code> .
<code>\ps</code>	Вставка деякого тексту після підпису - постскриптум.
<code>\\</code>	Ручне розбиття тексту листа на окремі рядки.

Приклад створення документу у стилі ділового листа:

```
% стиль документу - лист
\documentclass[12pt]{letter}
\usepackage[cp1251]{inputenc}
\usepackage[ukrainian]{babel}
% задає приватний стиль листа
\address{Б.Д.Василів, \\
    Прикарпатський національний Університет, \\
    Івано-Франківськ, \\Україна}
\signature{Василів Б.Д.}
\date{10 лютого 2012 року }
\telephone{59-60-58}
% відображається тільки при
% діловому стилі листа
\location{вул. Шевченка, 57}
% відображається тільки
% при діловому стилі листа
\begin{document}
% текст звернення
\begin{letter}

{Декану факультету\\
    математики та інформатики, \\
    доц. Пилипіву В.М}

%обов'язковий параметр - звернення до адресату
\opening{Шановний пане декан}
%початок самого листа
Я дуже надіюсь, що здам всі практичні
роботи на відмінно, але їх багато і я не встигаю.
У мене до вас велике прохання - прошу
дозволити мені здавати менше робіт.

\closing{Щиро ваш,}
% текст, що буде виведено після листа
%(постскриптум)
\ps{Буду вам щиро вдячний!!!}
\end{letter}
\end{document}
```

Результат компіляції документу:

Б.Д.Василів,
Прикарпатський національний Університет,
Івано-Франківськ,
Україна

10 лютого 2012 року

Декану факультету
математики та інформатики,
доц. Пилипіву В.М

Шановний пане декан

Я дуже надіюсь, що здам всі практичні роботи на відмінно, але їх багато і я не встигаю. У мене до вас велике прохання - прощу дозволити мені здавати менше робіт.

Щиро ваш.

Василів Б.Д.

Буду вам щиро вдячний!!!

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Особливу увагу зверніть на структуру **LaTeX** – документів у стилі статті та ділового листа, задання нумерації сторінок та їх стилю, налаштувань параметрів сторінки та можливостями розбиття документу на підрозділи.
2. Введіть приклади, скомпілюйте їх та перегляньте результат. При необхідності виправте помилки, що виникають при компіляції документу.

3. Взявши за основу приклад на сторінці 25, створіть документ у стилі статті відповідно до індивідуального завдання, де використайте наступні елементи оформлення:
 - титульну сторінку із вашими даними: прізвищем та номером групи;
 - до номера групи зробіть примітку, в якій задайте назву навчального закладу та факультет;
 - матеріал для статті –теоретичний матеріал підрозділів «Поля та розміри сторінок» і «Титульна сторінка, заголовок та анотація», які сформатуйте з використанням нумерованих та нелінійних розділів та підрозділів;
 - анотацію до документу задайте самостійно;
 - в документі використайте коментарі для пояснення ходу формування документу.
4. Створіть документ у стилі ділового листа, матеріал для якого придумайте самостійно. При його формуванні використати команди стилю **letter** із таб. №7.

Завдання для індивідуального виконання

Варіант	Параметри оформлення
1	2
1.	Стильові опції: 11pt, оформити документ у дві колонки, орієнтація сторінки альбомна. Стиль оформлення сторінки: номер сторінки вгорі. Поля та розмір сторінки: ширина тексту – 15 см, верхнє поле – 20 мм.
2.	Стильові опції: 12pt, розмір паперу legalpaper . Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – рядкові букви. Поля та розмір сторінки: ліве поле – 25 мм, висота тексту – 250 мм.
3.	Стильові опції: 10pt, розмір паперу legalpaper . Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – великі букви. Поля та розмір сторінки: ліве поле – 25 мм, верхнє поле – 25 мм.

1	2
4.	Стильові опції: 12pt, розмір паперу <i>a5paper</i> . Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 20 мм, верхнє поле – 30 мм.
5.	Стильові опції: 10pt, розмір паперу <i>legalpaper</i> . Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – малі букви. Поля та розмір сторінки: ліве поле – 15 мм, верхнє поле – 20 мм.
6.	Стильові опції: 12pt, орієнтація сторінки альбомна. Стиль оформлення сторінки: номер сторінки вгорі, тип нумерації – римські малі цифри. Поля та розмір сторінки: ширина тексту – 15 см, висота тексту – 550pt.
7.	Стильові опції: 10pt, оформити документ у дві колонки. Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 15 мм, верхнє поле – 35 мм.
8.	Стильові опції: 12pt, розмір паперу <i>a5paper</i> . Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – римські великі цифри. Поля та розмір сторінки: ширина тексту – 350pt, верхнє поле – 30мм.
9.	Стильові опції: 10pt. Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – великі букви. Поля та розмір сторінки: ліве поле – 2 см, верхнє поле – 20 мм.
10.	Стильові опції: 12pt, розмір паперу <i>legalpaper</i> . Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 25мм, висота тексту – 50 мм.
11.	Стильові опції: 10pt, розмір паперу <i>a5paper</i> . Стиль оформлення сторінки: номер сторінки в низу сторінки, тип нумерації – арабські цифри. Поля та розмір сторінки: ліве поле – 15 мм, верхнє поле – 25 мм.
12.	Стильові опції: 12pt, розмір паперу <i>legalpaper</i> . Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – римські малі цифри. Поля та розмір сторінки: ліве поле – 3 см, верхнє поле – 30 мм.

1	2
13.	Стильові опції: 10pt, орієнтація сторінки альбомна. Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ширина тексту – 550pt, висота тексту – 290pt.
14.	Стильові опції: 12pt, оформити документ у дві колонки. Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – арабські цифри. Поля та розмір сторінки: ліве поле – 25 мм, верхнє поле – 15 мм.
15.	Стильові опції: 10pt, розмір паперу <i>a5paper</i> . Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – малі букви. Поля та розмір сторінки: ліве поле – 2 см, верхнє поле – 25 мм.
16.	Стильові опції: 12pt, розмір паперу <i>legalpaper</i> . Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 3 см, верхнє поле – 2 см.
17.	Стильові опції: 10pt, розмір паперу <i>a5paper</i> . Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – римські великі цифри. Поля та розмір сторінки: ліве поле – 30 мм, висота тексту – 510pt.
18.	Стильові опції: 12pt, орієнтація сторінки альбомна. Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – арабські цифри. Поля та розмір сторінки: ліве поле – 25 мм, верхнє поле – 15 мм.
19.	Стильові опції: 10pt. Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 15 мм, верхнє поле – 50pt.
20.	Стильові опції: 12pt, розмір паперу <i>legalpaper</i> . Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – римські великі цифри. Поля та розмір сторінки: ліве поле – 2 см, верхнє поле – 30 мм.
21.	Стильові опції: 10pt, оформити документ у дві колонки. Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – римські малі цифри. Поля та розмір сторінки: ширина тексту – 15 см, верхнє поле – 60pt.
22.	Стильові опції: 12pt, орієнтація сторінки альбомна. Стиль оформлення сторінки: без номерів сторінок. Поля та розмір сторінки: ліве поле – 25 мм, верхнє поле – 10 мм.

1	2
23.	<p>Стильові опції: 10pt, оформити документ у дві колонки.</p> <p>Стиль оформлення сторінки: номер сторінки вгорі, тип нумерації – арабські цифри.</p> <p>Поля та розмір сторінки: ліве поле – 20 мм, висота тексту – 200 мм.</p>
24.	<p>Стильові опції: 12pt, розмір паперу <i>a5paper</i>.</p> <p>Стиль оформлення сторінки: номер сторінки внизу, тип нумерації – римські малі цифри.</p> <p>Поля та розмір сторінки: ліве поле – 25 мм, верхнє поле – 2 см.</p>
25.	<p>Стильові опції: 12pt, орієнтація сторінки альбомна.</p> <p>Стиль оформлення сторінки: без номерів сторінок.</p> <p>Поля та розмір сторінки: ліве поле – 3 см, верхнє поле – 3 см.</p>
26.	<p>Стильові опції: 11pt, розмір паперу <i>legalpaper</i>.</p> <p>Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – малі букви.</p> <p>Поля та розмір сторінки: ліве поле – 15 мм, верхнє поле – 30 мм.</p>
27.	<p>Стильові опції: 12pt, розмір паперу <i>a5paper</i>.</p> <p>Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – римські малі цифри.</p> <p>Поля та розмір сторінки: ширина тексту – 20 см, верхнє поле – 20 мм.</p>
28.	<p>Стильові опції: 11pt, орієнтація сторінки альбомна.</p> <p>Стиль оформлення сторінки: без номерів сторінок.</p> <p>Поля та розмір сторінки: ліве поле – 20 мм, верхнє поле – 20 мм.</p>
29.	<p>Стильові опції: 10pt, розмір паперу <i>a5paper</i>.</p> <p>Стиль оформлення сторінки: номер сторінки знизу, тип нумерації – римські великі цифри.</p> <p>Поля та розмір сторінки: ліве поле – 20 мм, верхнє поле – 1,5 см.</p>
30.	<p>Стильові опції: 12pt, орієнтація сторінки альбомна.</p> <p>Стиль оформлення сторінки: номер сторінки зверху, тип нумерації – арабські цифри.</p> <p>Поля та розмір сторінки: ліве поле – 2 см, верхнє поле – 55pt.</p>

Контрольні питання

1. Які стилі документів ви знаєте?
2. Як створити документ у стилі статті?
3. Як створити титульну сторінку до статті?
4. Як створити титульну сторінку в ручному режимі?
5. Як розмістити текст документу у дві колонки?
6. Перерахуйте основні команди формування документу в стилі «лист».
7. Які атрибути та параметри, що управляють розміром та положенням тексту на сторінці, ви знаєте?
8. Перерахуйте основні команди розбиття тексту на підрозділи.
9. Яким чином здійснюється нумерація розділів та підрозділів в документі?
10. Як виключити нумерацію підрозділу?
11. Як включити нумерацію сторінок?
12. Як задати стиль оформлення номерів сторінок?

Лабораторна робота №3

Тема роботи: Робота із текстом та абзацами у видавничій системі LaTeX

Мета роботи: формування навичок роботи із абзацами: розриви рядків та сторінок, переноси слів, вертикальні відступи та оточення для форматування тексту.

Для виконання роботи необхідно знати:

- способи вводу та редагування простого тексту;
- управління перенесенням слів на інший рядок;
- ввід розділових знаків;
- управління розривами рядків;
- управління міжрядковими інтервалами;
- розриви сторінок;
- основні оточення для форматування тексту.

Теоретичні відомості

Ввід простого тексту

У вихідному тексті слова та інші допустимі символи повинні бути розділені між собою одним чи декількома пробілами (їх кількість на етапі компіляції значення не має). Перехід на новий рядок теж розглядається як пробіл. Управління величиною пробілу між словами здійснюється за допомогою команд, приведених в таб. №8.

Таблиця №8. Команди управління величиною пробілу між словами в тексті документу

Приклад команди	Результат компіляції	Коментар
1	2	3
Величина пробілу	Величина пробілу	Величина пробілу після компіляції може мінятися
Величина~пробілу	Величина пробілу	Нерозривний пробіл фіксованої ширини
Величина\ пробілу	Величина пробілу	Пробіл фіксованої ширини

1	2	3
Величина \,пробілу	Величина пробілу	Пробіл рівний ширині стандартного символу
Величина \enskip{} пробілу	Величина пробілу	Пробіл рівний ширині стандартного символу
Величина \quad{} пробілу	Величина пробілу	Пробіл рівний подвійній ширині стандартного символу
Величина \qqquad{} пробілу	Величина пробілу	Пробіл рівний 4 ширинам стандартного символу
Величина \hspace{16mm}пробілу	Величина пробілу	Величина пробілу задається значенням обов'язкового параметру у фігурних дужках. В даному випадку – 16мм.
Величина \hspace{0mm}пробілу	Величина пробілу	Слова без пробілу – пробіл між словами відсутній.
Величина \hspace{-6mm}пробілу	Величина пробілу	Від'ємне значення параметру веде до зменшення пробілу на вказану величину (накладання двох слів)

Примітка: Після таких команд повинен стояти тільки один пробіл. Якщо їх стоїть більше, то у скомпільованому документі пробіл між словами буде збільшуватися.

Розділові знаки

Крапки, коми, знаки оклику та знаки питання у тексті документу використовують як звичайні розділові знаки. Вони ставляться впритул до попереднього слова, а після них обов'язково слідує пробіл.

В документі можуть використовуватися три різні типи тире (див. таб. №9):

- одинарне тире (дефіс) як з'єднувальний знак у складних словах. Ніколи не відділяється пробілами від попереднього та наступного слова;
- подвійне (коротке) тире як знак границі від і до;
- потрійне (звичайне) тире. Заміняє собою пропущене слово.

Таблиця №9. Використання різних типів тире

Тип тире	Команда	Результат компіляції
Дефіс	інженер-програміст	інженер-програміст
Коротке тире	сторінки 15--24	сторінки 15-24
Звичайне тире	це --- звичайне тире	це—звичайне тире

У таб. №10 та №11 наведено приклади правильного та неправильного формування лапок та трикрапок у документі **LaTeX**.

Таблиця №10. Вставка лапок у документ

Команда	Результат компіляції	Примітка
"це --- звичайне тире"	"це—звичайне тире"	Не правильно
``це --- звичайне тире``	"це—звичайне тире"	Правильно

Таблиця №11. Вставка у документ трьох крапок

Команда	Результат компіляції	Примітка
це три крапки...	це три крапки...	Не правильно
це три крапки <code>\dots</code>	це три крапки...	Правильно

Розбиття документу на абзаци

Для розбиття тексту на окремі абзаци достатньо розділити його у відповідних місцях одним або кількома порожніми рядками (значення немає). Розбиття тексту на абзаци можна також задати за допомогою команди `\par`. Щоби рядки тексту мали однакову довжину, система **LaTeX** сама розбиває текст по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами використовується перенесення слів за правилами української мови (якщо на початку документу підключено пакет `\usepackage[ukrainian]{babel}`). Величина абзацного відс-

тупу формується системою **LaTeX**, а при необхідності може бути змінена у преамбулі документу командою `\setlength{\parindent}{\text{Відступ}}`. Величина відступу задається у будь-яких допустимих одиницях вимірювання.

Для початку нового абзацу без абзацного відступу служить команда `\noindent`, а для виступу першого рядка абзацу додатково використовується команда `\hangindent`.

Управління переносом слів

Способом переносу слів можна управляти за допомогою команди `\rightthyphenmin` згідно наступного синтаксису (у приведені нижче прикладі параметр дозволяє переносити із рядка на рядок не менше 4 символів; по замовчужанню дозволено переносити не менше 3 символів):

`\rightthyphenmin=4`

Для одноразового дозволу переносу слова в даному місці передбачена команда `\-`, яка ставиться у місці бажаного переносу (навіть якщо це суперечить граматиці мови). Наприклад `{ма\ -тема\ -тика}`.

Для задання способу переносу слова в межах всього документу необхідно скористатися командою `\hyphenation` (в преамбулі, або тілі документу). Наприклад:

- команда `\hyphenation{ма\ -тема\ -тика прих\ -лад}` задасть спосіб переносу слова у межах всього документу;
- команда `\hyphenation{математика}` заборонить перенос слова у межах всього документу.

В якості аргументу команди може бути задано кілька слів, розділених пропущками.

Інколи виникає ситуація, коли текст не поміщається в дозволені межах навіть із використанням переносів слів (відбувається переповнення рядка). Для коректного управління такого роду ситуаціями служать наступні дві команди:

`\sloppy` дозволяє «ослабити» стандарти системи по формуванню рядків за рахунок збільшення інтервалів між словами – як наслідок пропуски між словами можуть бути надто великими;

`\fussy` повертає звичайний режим форматування тексту. Як альтернатива можна використовувати команду `\sloppy` в середині групи, що задає область її дії: по виході із групи даний режим форматування документу буде припинено.

Управління розривами рядків

Розбиття тексту на окремі рядки система здійснює самостійно. Для ручного управління цим процесом служать наступні команди:

- \ перед пробілом вказує на те, що при форматуванні він не може бути збільшений (пробіл фіксованої ширини);
- ~ нерозривний пробіл: команда задає, в якому місці новий рядок починати не можна. Наприклад **теорема~1**;
- \mbox заборона перенесення заданого як аргумент слова чи словосполучення у даному місці. Наприклад **\mbox{теорема 1}**;
- \\ примусовий розрив рядка без розтягнення тексту на всю його ширину;
- \@ перед крапкою вказує на те, що вона закінчує речення;
- \linebreak примусовий розрив рядка із вирівнюванням тексту по всій ширині сторінки.

Примітка: як необов'язковий параметр у квадратних дужках можна задати величину вертикального відступу після розриву рядка. Наприклад:

Приклад 1, що ілюструє примусовий розрив рядка \\[3mm] та встановлення величини вертикального відступу у 3мм

Приклад 2, що ілюструє примусовий розрив рядка %текст буде вирівняно по ширині сторінки \linebreak без встановлення величини вертикального відступу

Результат компіляції:

Приклад 1, що ілюструє примусовий розрив рядка

та встановлення величини вертикального відступу у 3мм

Приклад 2, що ілюструє примусовий розрив рядка без встановлення величини вертикального відступу

Вертикальні відступи у рядках

Відстанню між абзацами можна управляти за допомогою команд (у порядку збільшення відстані): **\smallskip**, **\medskip**, **\bogskip** (команду розміщують або після тексту абзаци, або безпосередньо після кінця абзацу). Команда **\vspace{}** дозволяє задати величину відступу між абзацами (обов'язковий параметр – величини міжабзацного відступу).

Команди діють тільки в межах даного абзацу, а для задання величини відступу в межах всього документу необхідно в преамбулі документу скористатися командою **\parskip** із обов'язковим параметром – величиною міжабзацного відступу.

Примітка: значення обов'язкових параметрів можуть бути і від'ємними: в цьому випадку міжабзацний відступ буде зменшено.

Міжрядковий інтервал

Інтервал між рядками формується системою самостійно виходячи зі стилю документу. Для прямого управління інтервалом служать наступні команди:

- **\linespread**: ставиться у преамбулі документу із обов'язковим параметром – величиною міжрядкового інтервалу. Стосується всього документу. Наприклад **\linespread{2.0}** задає подвійний міжрядковий інтервал;
- **\renewcommand{\baselinestretch}{x}** дозволяє поміняти міжрядковий інтервал в межах групи. **x** задає відношення нової величини міжрядкового інтервалу до стандартної величини, що визначається стилем документу. Наприклад при **x=1.2** відстань між рядками збільшиться в 1.2 рази.

Розриви сторінок

Розбиття документу на сторінки формується виходячи із оптимального з точки зору видавничої системи. Для явного управління процесом можна використати наступні команди:

- **\nopagebreak** заборона розриву сторінки в межах даного абзацу;
- **\newpage** або **\pagebreak** - примусовий розрив документу. Сторінка до кінця залишиться порожньою;
- **\pagebreak \mbox{}** **\pagebreak** – створення порожньої сторінки. Для друкування порожньої сторінки необхідно вставити на неї деякий недрукований символ командою **\mbox{}**, бо інакше система сторінку не відобразить.

Оточення для управління розміщенням тексту у абзацах

Оточення **quote** дозволяє формувати вибраний фрагмент документу із додатковими відступами від країв документу (для цитат, важливих фраз чи прикладів). Сам текст буде відділений від основного документу порожніми рядками. Наприклад:

Приклад використання оточення (`\bf quote`)

`\begin{quote}`

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому.

`\end{quote}`

Для зменшення інтервалів між словами

`\LaTeX` використовує перенесення слів за правилами української мови

абзацу в цілому. Для зменшення інтервалів між словами `\LaTeX` використовує перенос слів за правилами української мови.

`\end{center}`

Результат компіляції:

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами `\TeX` використовує перенос слів за правилами української мови.

Результат компіляції:

Приклад використання оточення `quote`

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому.

Для зменшення інтервалів між словами `\TeX` використовує перенесення слів за правилами української мови

Оточення `quotation` дозволяє формувати вибраний фрагмент документу із додатковими відступами від країв документу та формуванням абзацних відступів. Використовується для формування довгих цитат, що складаються із кількох абзаців.

Оточення `center` розміщує текст по центру рядків. Наприклад:

`\begin{center}`

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування

Команда `\centerline` центрує текст в межах одного рядка. Наприклад:

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками.

`\centerline{Щоби рядки тексту мали однакову довжину, }`

система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами `\LaTeX` використовує перенесення слів за правилами української мови.

Результат компіляції:

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками.

Щоби рядки тексту мали однакову довжину.

система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами `\TeX` використовує перенесення слів за правилами української мови.

Оточення *flushleft* вирівнює текст абзацу по лівому краю. Наприклад:

```
\begin{flushleft}
```

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками.

Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами *LaTeX* використовує перенос слів за правилами української мови.

```
\end{flushleft}
```

Результат компіляції:

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками.

Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому. Для зменшення інтервалів між словами *LaTeX* використовує перенос слів за правилами української мови.

Оточення *flushright* вирівнює текст по правому краю сторінки.

Оточення *verbatim* дозволяє відобразити текст буквально так, як він набраний у вихідному документі: із всіма пробілами та розбиттям тексту по рядках. Аналогічний результат можна отримати за допомогою команди *\verb* (в якості символів-обмежувачів в команді можна використовувати будь-який символ, але не букви чи пробіли. Наприклад:

```
\begin{verbatim}
```

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками.

Щоби рядки тексту мали однакову довжину, система сама

формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації

формування абзацу в цілому.

```
\end{verbatim}
```

\verb+Для зменшення інтервалів між словами+
LaTeX використовує перенос слів за правилами української мови.

Результат компіляції:

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому.

Для зменшення інтервалів між словами *LaTeX* використовує перенос слів за правилами української мови.

Приклад використання символу * у оточенні *verbatim*:

```
\begin{verbatim*}
```

Для розбиття тексту на окремі абзаци достатньо розділити текст у відповідних місцях порожніми рядками. Щоби рядки тексту мали однакову довжину, система сама формує розбиття тексту по рядках та використовує додаткові пробіли між словами для оптимізації форматування абзацу в цілому.

```
\end{verbatim*}
```

*\verb**+Для зменшення інтервалів між словами +

LaTeX

використовує перенесення слів за правилами української мови.

Результат компіляції:

Для розбиття тексту на окремі абзаци
достатньо розділити текст у відповідних
місцях порожніми рядками. Щоби рядки тексту
мали однакову довжину, система сама формує
розбиття тексту по рядках та використовує
додаткові пробіли між словами для
оптимізації форматування абзацу в цілому.

Для зменшення інтервалів між словами LaTeX використовує
перенесення слів за правилами української мови

Для вставки в документ виноски служить команда `\footnote`. Її
обов'язковим параметром є текст, який буде виведено у виносці. Текст
виноски може складатися з кількох абзацив: в цьому випадку вони,
відділяючись порожніми рядками. В документі виноска має наскрізну
нумерацію. Наприклад:

```
\footnote{це є приклад використання команди verb*}  
\verb*+Для зменшення інтервалів між  
словами+  
LaTeX використовує перенесення  
слів за правилами української мови
```

Результат компіляції:

¹ Для зменшення інтервалів між словами LaTeX використовує
перенесення слів за правилами української мови

¹це є приклад використання команди verb*

У LaTeX не передбачена можливість заново починати нумерацію
виносок на кожній сторінці. Варіантом організації нової нумерації вино-
сок на сторінці є використання команди `\footnote` з необов'язковим
аргументом. Номер виноска розміщується у квадратних дужках перед
обов'язковим параметром. Наприклад:

`\footnote[111]{Виноска`

Ілюструє управління номером виноска}
Приклад формування виноска у документі.

`\footnote[222]{Виноска до другого фрагменту
документу}`

У документі нумерація виноска є наскрізною.
Для управління номером виноска необхідно задати
в квадратних дужках номер виноска.
Сама виноска може бути сформована у
декілька рядків

Результат компіляції:

¹¹¹ Приклад формування виноска у документі.

²²² У документі нумерація виноска є наскрізною. Для управління но-
мером виноска необхідно задати в квадратних дужках номер виноска.
Сама виноска може бути сформована у декілька рядків

¹¹¹Виноска

Ілюструє управління номером виноска

²²²Виноска до другого фрагменту документу

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте їх та перегляньте результат. При необхідності виправте помилки, що виникають при компіляції.
2. Створіть документ у стилі статті, сформуйте його відповідно до індивідуального завдання. Застосуйте також наступні елементи оформлення:
 - Титульну сторінку із назвою статті та вашими даними: прізвищем та номером групи;
 - розмір шрифту – 11pt;
 - в титульному листі задайте дату створення документу явним чином;
 - матеріал документу почніть із нової сторінки.

Матеріал для створення документу

Підготовка статті у видавничій системі LaTeX

На відміну від більшості текстових процесорів, які дозволяють в момент набору тексту переглянути на екрані дисплею результат у тому вигляді, яким він буде надрукований на папері, LaTeX - це система логічного проектування тексту.

Процес отримання друкованого документа в LaTeX складається з двох етапів. Перший - це створення вхідного файлу у форматі LaTeX з розширенням `.tex`. `File.tex` містить як сам текст документу, так і команди стилю й форматування для верстки друкованого матеріалу. Створювати `tex`-файл можна практично в будь-якому текстовому редакторі, так як LaTeX-файл - це звичайний текстовий файл стандарту ASCII (American Standard Code for Information Interchange).

Існують також і спеціально розроблені текстові редактори для набору LaTeX-івських файлів. Так LaTeX тісно інтегрований з текстовим редактором WinEdit. Останній має потужний редакторський сервіс, який суттєво спрощує роботу по набору математичних формул, компіляції `tex`-файлів.

В більшості випадків форматування тексту зводиться до вживання команд, що визначають стиль всього документу, чи окремого його фрагменту. Задача полегшується тим, що LaTeX містить готові макети професійних стилів, які підключає компілятор LaTeX при виготовленні верстки.

Другий етап - компілювання `file.tex` у зверстаний файл з розширенням `.dvi`, який готовий для читання чи друку засобами LaTeX.

Основні поняття системи:

Команди в тексті вхідного файлу завжди починаються з символу `\`, за яким може слідувати один символ, що не є літерою (наприклад `{`, `%`, `$`, тощо), або ж послідовність літер, що є іменем команди. Кожна команда обов'язково закінчується пробілом або символом-нелітерою. В LaTeX деякі команди закінчуються символом `*`, що робить команди гнучкішими. Наприклад, команда `\section` створює новий розділ документа. Команда `\section*` також створює новий розділ, але не нумерує його. LaTeX розрізняє великі і малі літери в іменах команд.

Група - важливе поняття TEX-у. Група складається з двох символів `{}`, всередині яких може бути текст, або команди. Команди, якщо вони розміщені всередині групи, діють тільки на текст групи, що розміщений після команди. Групи можуть бути вкладеними одна в одну. Фігурні дужки, що використовуються для позначення групи, повинні бути

їбалансованими в тексті вхідного файлу: число відкриваючих та закриваючих дужок повинно бути однаковим.

Оточення - фрагмент файлу, який починається з команди `\begin{Im'я оточення}` і закінчується `\end{Im'я оточення}`. Частина тексту, яка знаходиться всередині оточення, утворює групу.

Продовження слідує...

Завдання для індивідуального виконання

Варіант	Завдання для виконання
1	2
1.	Здайте величину абзацного відступу для всього документу 2 см; 1-ший абзац вирівняйте по правому краю сторінки; 2-гий абзац вирівняйте по центру сторінки; 3-тій абзац вирівняйте по лівому краю сторінки; в 4-тому абзаці використайте команди управління пробілами; відстань між 5-тим та 6-тим абзацами задайте рівною 30 pt; до 6-го абзацу задайте виноску. Виноску сформуйте у два рядки, а матеріал задайте самостійно. Номер виноски -16; для 7-го абзацу збільшіть міжрядковий інтервал в 1.5 рази; 7-мий абзац почніть із нової сторінки.
2.	Здайте величину абзацного відступу для першого абзацу 2.5 см; 2-гий абзац відобразіть буквально; 4-тий абзац вирівняйте по центру сторінки; 6-тий абзац вирівняйте по правому краю сторінки; в 7-му абзаці використайте команди управління пробілами; відстань між 1-шим та 2-гим абзацами задайте рівною 35 pt; до 4-го абзацу створіть виноску. Матеріал виноски задайте самостійно; для 8-го абзацу зменшіть міжрядковий інтервал в 1.1 рази; 6-мий абзац почніть із нової сторінки.
3.	Здайте величину абзацного відступу для всього документу 25 pt; 1-ший абзац відобразіть буквально із пробілами; 2-гий абзац вирівняйте по правому краю сторінки; в 3-тому абзаці використайте команди управління пробілами; відстань між абзацами задайте рівною 25 pt; до 3-го абзацу створіть три виноски. Матеріал виноски задайте самостійно. Номери виносок: 6,9,10; для 5-го абзацу зменшіть міжрядковий інтервал в 1.2 рази; 7-мий абзац почніть із нової сторінки.

1	2
4.	<p>Задайте величину абзацного відступу для всього документу 1.5 см;</p> <p>1-ший абзац вирівняйте по центру сторінки;</p> <p>2-гий абзац відобразіть буквально;</p> <p>4-тій абзац вирівняйте по правому краю сторінки;</p> <p>в 5-тому абзаці використайте команди управління пробілами;</p> <p>відстань між 3-тім та 4-тим абзацами задайте рівною 2 см ;</p> <p>до 7-го абзацу створіть дві виноски. Виноски сформуйте у два рядки, а матеріал задайте самостійно. Номери виносок: 6 та 7;</p> <p>для 3-го абзацу збільшіть міжрядковий інтервал в 1.3 рази;</p> <p>3-тій та 6-тий абзаци почніть із нової сторінки.</p>
5.	<p>Задайте величину абзацного відступу для перших 3-х абзацив 25 pt;</p> <p>1-гий абзац вирівняйте по центру сторінки;</p> <p>2-гий абзац відобразіть буквально із пробілами;</p> <p>3-тій абзац вирівняйте по правому краю сторінки;</p> <p>в 5-тому абзаці використайте команди управління пробілами;</p> <p>відстань між 1-шим та 2-гим абзацами задайте рівною 30 pt;</p> <p>до 7-го абзацу створіть виноску. Виноску сформуйте у два рядки, а матеріал задайте самостійно;</p> <p>для 6-го абзацу збільшіть міжрядковий інтервал в 1.5 рази;</p> <p>5,7-мий абзаци почніть із нової сторінки.</p>
6.	<p>Задайте величину абзацного відступу для всього документу 20 pt;</p> <p>1-ший абзац вирівняйте по правому краю сторінки. Для кількох слів абзацу задайте заборону переносу;</p> <p>2-гий абзац оформіть із додатковими відступами від країв;</p> <p>4-тій абзац вирівняйте по лівому краю сторінки;</p> <p>в 3-тьому абзаці використайте команди управління пробілами;</p> <p>відстань між 2-гим та 3-тім абзацами задайте рівною 20 pt;</p> <p>до 6-го абзацу створіть три виноски. Матеріал виноски задайте самостійно. Номери виносок: 1,6,7.</p> <p>для 5-го абзацу зменшіть міжрядковий інтервал в 1.1 рази;</p> <p>2, 4,7-мий абзаци почніть із нової сторінки.</p>
7.	<p>Задайте величину абзацного відступу для абзацив 3,4,5 1.5 см;</p> <p>1-ший абзац оформіть із додатковими відступами від країв;</p> <p>2-гий абзац вирівняйте по центру сторінки;</p> <p>3-тій абзац вирівняйте по правому краю сторінки. Для кількох слів абзацу задайте заборону переносу;</p> <p>в 4-тому абзаці використайте команди управління пробілами між словами;</p> <p>відстань між 4-тим, 5-тим та 6-тим абзацами задайте рівною 2 см;</p> <p>до 7-го абзацу створіть виноску. Виноску сформуйте у два рядки, а матеріал задайте самостійно.</p> <p>для 5-го абзацу збільшіть міжрядковий інтервал в 1.2 рази;</p> <p>2-гий та 7-мий абзаци почніть із нової сторінки.</p>

1	2
8.	<p>Задайте величину абзацного відступу для всього документу 25 pt;</p> <p>1-ший абзац вирівняйте по лівому краю сторінки. Для кількох слів абзацу задайте заборону переносу;</p> <p>2-гий абзац вирівняйте по центру сторінки;</p> <p>3-тій абзац оформіть із додатковими відступами від країв;</p> <p>в 5-тому абзаці використайте команди управління пробілами між словами;</p> <p>відстань між 3-тім, 4-тим та 5-тим абзацами задайте рівною 20 pt;</p> <p>до 7-го абзацу створіть дві виноски. Виноски сформуйте у два рядки, а матеріал задайте самостійно.</p> <p>для 6-го абзацу збільшіть міжрядковий інтервал в 1.2 рази;</p> <p>3-тій та 7-мий абзаци почніть із нової сторінки.</p>
9.	<p>Задайте величину абзацного відступу для всього документу 2.5 см;</p> <p>1-ший абзац вирівняний по лівому краю сторінки;</p> <p>2-гий абзац оформіть із додатковими відступами від країв;</p> <p>3-тій абзац відобразіть буквально із пробілами;</p> <p>в 4-тому абзаці для кількох слів задайте заборону переносу;</p> <p>в 5-тому абзаці використайте команди управління пробілами;</p> <p>відстань між 4-тим та 5-тим абзацами задайте рівною 10 pt;</p> <p>до 1-го абзацу створіть виноску. Виноску сформуйте у два рядки, а матеріал задайте самостійно. Номер виноски: 6.</p> <p>для 6-го абзацу збільшіть міжрядковий інтервал в 1.2 рази;</p> <p>3-тій та 4-тій абзаци почніть із нової сторінки.</p>
10.	<p>Задайте для всього документу подвійний інтервал;</p> <p>1-ший абзац оформіть із додатковими відступами від країв та абзацними відступами;</p> <p>2-гий абзац вирівняйте по центру сторінки;</p> <p>3-тій абзац вирівняйте по правому краю сторінки;</p> <p>в 5-тому абзаці використайте команди управління пробілами;</p> <p>відстань між 1-шим, 2-гим та 3-тім абзацами задайте рівною 20 pt;</p> <p>до 4-го абзацу створіть виноску. Матеріал виноски задайте самостійно. Номер виноски: 3;</p> <p>для 6-го абзацу збільшіть міжрядковий інтервал в 1.3 рази;</p> <p>2-гий та 5-тий абзаци почніть із нової сторінки.</p>
11.	<p>Задайте величину абзацного відступу для всього документу 2 см;</p> <p>1-ший абзац відобразіть буквально;</p> <p>2-тій абзац вирівняйте по лівому краю сторінки;</p> <p>для 3-тього абзацу задайте подвійний інтервал;</p> <p>відстань між 4-тим та 5-тим абзацами задайте рівною 2 см;</p> <p>в 5-тому абзаці використайте команди управління переносами слів;</p> <p>до 6-го абзацу створіть три виноски. Матеріал виносок задайте самостійно;</p> <p>для 8-го абзацу зменшіть міжрядковий інтервал в 1.1 рази;</p> <p>2-гий, 4-тій та 6-тий абзаци почніть із нової сторінки.</p>

1	2
12.	<p>Задайте для всього документу подвійний інтервал; для 1-шого абзацу задайте величину абзацного відступу 3 см; 2-гий абзац вирівняйте по правому краю сторінки; 3-гий абзац вирівняйте по центру сторінки; 4-тий абзац відобразіть буквально із пробілами; в 5-тому абзаці використайте команди управління пробілами; відстань між 1-шим, 2-гим та 3-тім абзацами задайте рівною 2 см; до 3-го абзацу створіть дві виноски. Виноски сформууйте у два рядки, а матеріал задайте самостійно; для 6-го абзацу збільшіть міжрядковий інтервал в 1.1 рази; 7-мий абзац почніть із нової сторінки.</p>
13.	<p>Задайте величину абзацного відступу для всього документу 2 см; задайте для 1-шого та 4-того абзаців подвійний інтервал; 1-ший абзац вирівняйте по правому краю поля сторінки; 2-гий абзац вирівняйте по центру поля сторінки; 3-тий абзац оформіть із додатковими відступами від країв; в 1-шому абзаці використайте команди управління переносами слів; в 2-тому абзаці використайте команди управління пробілами; відстань між 1-шим та 2-тим абзацами задайте рівною 2см; до 4-го абзацу створіть виноску. Матеріал виноски задайте самостійно; для 6-го абзацу зменшіть міжрядковий інтервал в 1.2 рази; 5-тий абзац почніть із нової сторінки.</p>
14.	<p>Задайте величину абзацного відступу для всього документу 2 см; 1-ший абзац вирівняйте по правому краю сторінки; 2-гий абзац вирівняйте по центру поля сторінки; 3-тий абзац вирівняйте по лівому краю сторінки; в 4-тому абзаці використайте команди управління пробілами; відстань між 5-тим та 6-тим абзацами задайте рівною 30 pt; до 6-го абзацу створіть виноску. Виноску сформууйте у два рядки, а матеріал задайте самостійно. Номер виноски – 16; для 7-го абзацу збільшіть міжрядковий інтервал в 1.5 рази; 6-тий абзац почніть із нової сторінки.</p>
15.	<p>Задайте величину абзацного відступу для 1-шого абзацу 1.5 см; 1-ший абзац вирівняйте по правому краю сторінки; 2-гий абзац відобразіть буквально; 3-тий абзац оформіть із додатковими відступами від країв; в 4-тому абзаці використайте команди управління переносами; відстань між 1-шим, 2-гим та 3-тім абзацами задайте рівною 2см; до 5-го абзацу створіть три виноски. Виноски сформууйте у два рядки, а матеріал задайте самостійно. Номери виносок: 1,2 та 5; для 6-го абзацу збільшіть міжрядковий інтервал в 1.3 рази; 2-гий та 6-тий абзаци почніть із нової сторінки.</p>

1	2
16.	<p>Задайте величину абзацного відступу для всього документу 17 pt; 1-ший абзац вирівняйте по лівому краю сторінки; 2-гий абзац оформіть із додатковими відступами від країв; 3-тий абзац вирівняйте по центру сторінки. Для кількох слів абзацу задайте заборону переносу; в 5-тому абзаці використайте команди управління пробілами; відстань між 3-тім та 4-тим абзацами задайте рівною 30 pt; до 6-го абзацу створіть дві виноски. Матеріал виноски задайте самостійно. Номери виносок: 3,7. для 5-го абзацу збільшіть міжрядковий інтервал в 1.3 рази; 4,6-тий абзаци почніть із нової сторінки.</p>
17.	<p>Задайте величину абзацного відступу для всього документу 1.5 см; 2-гий абзац відобразіть буквально; 1-ший абзац вирівняйте по правому краю сторінки; для 3-тього абзацу задайте подвійний інтервал; відстань між 4-тим та 5-тим абзацами задайте рівною 2 см; в 6-тому абзаці використайте команди управління переносами слів; до 6-го абзацу створіть виноску. Матеріал виносок задайте самостійно; для 7-го абзацу зменшіть міжрядковий інтервал в 1.3 рази; 4-тий та 8-мий абзаци почніть із нової сторінки.</p>
18.	<p>Задайте величину абзацного відступу для всього документу 2.5 см; 1-ший абзац відобразіть буквально; 2-гий абзац вирівняйте по правому центру; в 3-тому абзаці використайте команди управління пробілами та переносами слів; відстань між абзацами задайте рівною 20 pt; до 5-го абзацу створіть дві виноски (одна з них у два рядки). Матеріал винос- ки задайте самостійно. Номери виносок: 2,3; для 6-го абзацу збільшіть міжрядковий інтервал в 1.3 рази; 5-тий та 8-мий абзаци почніть із нової сторінки.</p>
19.	<p>Задайте величину абзацного відступу для всього документу 22 pt; 1-ший абзац вирівняйте по лівому краю сторінки; 2-гий абзац вирівняйте по правому краю сторінки; 3-тий абзац вирівняйте по центру сторінки; в 4-тому абзаці використайте команди управління переносами слів; відстань між 2-гим та 2-тім абзацами задайте рівною 2 см; до 6-го абзацу створіть виноску. Виноску сформууйте у два рядки, а матеріал задайте самостійно; для 7-го абзацу зменшіть міжрядковий інтервал в 1.2 рази; 3-тий та 7-мий абзаци почніть із нової сторінки.</p>

1	2
20.	<p>Задайте величину абзацного відступу для 1-шого абзацу 21 рт; 1-ший абзац вирівняйте по центру сторінки; 2-гий абзац відобразіть буквально; 3-тій абзац оформіть із додатковими відступами від країв та абзацними відступами; в 4-тому абзаці використайте команди управління величиною пробілу між словами; відстань між 3-тім, та 4-тим абзацами задайте рівною 20 рт; до 6-го абзацу створіть дві виноски. Матеріал виноски задайте самостійно. Номери виносок: 4 та 7; для 7-го абзацу зменшіть міжрядковий інтервал в 1.2 рази; 5-тий та 8-мий абзаци почніть із нової сторінки.</p>

Контрольні питання

1. Як задати нерозривний пробіл між словами?
2. Як задати пробіл фіксованої ширини?
3. Як задати пробіл заданої величини?
4. Як правильно вставити лапки в документ?
5. Перерахуйте основні типи тире в документі.
6. Як вставити в документ три крапки?
7. Перерахуйте команди управління розривами рядків.
8. Перерахуйте команди управління розривом сторінок.
9. Які оточення та команди вирівнювання тексту в межах абзацу ви знаєте?
10. Як поміняти міжрядковий інтервал?
11. Як розбити документ на сторінки?
12. Перерахуйте основні оточення для форматування тексту в документі.
13. Як розмістити текст по центру сторінки?
14. Яке призначення оточення *verbatim*?
15. Як вставити виноску в документ?
16. Як можна управляти номером виноски в документі?

Лабораторна робота №4

Тема роботи: Робота зі списками у видавничій системі LaTeX

Мета роботи: формування навичок роботи із нумерованими та маркованими списками: управління типом маркера та нумерацією; створення списку літератури та генерування посилань

Для виконання роботи необхідно знати:

- команди формування маркованого списку;
- команди формування нумерованого списку;
- команди формування списку літератури;
- генерування посилань на літературу та їх використання.

Теоретичні відомості

Створення маркованого списку

Марковані списки формуються за допомогою оточення *itemize*, яке дозволяє створювати списки максимум до чотирьох рівнів вкладення. Кожен елемент списку починається командою *\item*. Приклад формування маркованого списку із чотирма рівнями вкладення:

```

\begin{itemize}
  \item Це перший рівень вкладення
    \begin{itemize}
      \item Це другий рівень вкладення
        \begin{itemize}
          \item Це третій рівень вкладення
            \begin{itemize}
              \item Це четвертий рівень вкладення
            \end{itemize}
          \end{itemize}
        \end{itemize}
      \end{itemize}
    \end{itemize}
  \item Це другий запис першого рівня
    \begin{itemize}
      \item Це другий рівень вкладення
    \end{itemize}
  \item Це третій запис першого рівня
\end{itemize}
\begin{itemize}
  \item Це перший рівень вкладення

```

```

\begin{itemize}
  \item Це другий рівень вкладення
\begin{itemize}
  \item Це третій рівень вкладення
\begin{itemize}
  \item Це четвертий рівень вкладення
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}

```

Результат компіляції:

- Це перший рівень вкладення
 - Це другий рівень вкладення
 - * Це третій рівень вкладення
 - Це четвертий рівень вкладення
- Це другий запис першого рівня
 - Це другий рівень вкладення
- Це третій запис першого рівня
- Це перший рівень вкладення
 - Це другий рівень вкладення
 - * Це третій рівень вкладення
 - Це четвертий рівень вкладення

Якщо вигляд маркера (залежить від рівня вкладення), який використовується по замовчуванню (див попередній приклад) не підходить, то його можна задати явним чином за допомогою необов'язкового параметру до `\item`:

```

\begin{itemize}
  \item[-] Це перший рівень вкладення
  \begin{itemize}
    \item[=] Це другий рівень вкладення
  \begin{itemize}
    \item[q] Це третій рівень вкладення
  \begin{itemize}
    \item[+] Це четвертий рівень вкладення
  \end{itemize}
  \end{itemize}
  \end{itemize}
\end{itemize}

```

```

\end{itemize}
\end{itemize}
\end{itemize}
\item[-] Це другий запис першого рівня
\begin{itemize}
  \item[=] Це другий рівень вкладення
\end{itemize}
\item[-] Це третій запис першого рівня
\end{itemize}

```

Результат компіляції:

- Це перший рівень вкладення
 - = Це другий рівень вкладення
 - q Це третій рівень вкладення
 - + Це четвертий рівень вкладення
- Це другий запис першого рівня
 - Це другий рівень вкладення
- Це третій запис першого рівня

Вигляд маркерів, що використовуються за замовчуванням, можна поміняти в межах всього документа в преамбулі командою:

```
\renewcommand{рівень}{символ}
```

тут:

- **рівень**: рівень вкладення маркованого списку. Може приймати одне із наступних значень: `\labelitemi` (перший рівень вкладення); `\labelitemii` (другий рівень вкладення); `\labelitemiii` (третій рівень вкладення); `\labelitemiv` (четвертий рівень вкладення);
- **символ**: символ, що буде використовуватися для відповідного рівня вкладення.

Наприклад:

```

\documentclass[12pt]{article}
\usepackage{cp1251}{inputenc}
\usepackage[english,ukrainian]{babel}
\renewcommand{\labelitemi}{*}
\renewcommand{\labelitemii}{**}
\renewcommand{\labelitemiii}{***}

```

```

\renewcommand{\labelitemiv}{****}
\begin{document}
\begin{itemize}
  \item Це перший рівень вкладення
  \begin{itemize}
    \item Це другий рівень вкладення
    \begin{itemize}
      \item Це третій рівень вкладення
      \begin{itemize}
        \item Це четвертий рівень вкладення
      \end{itemize}
    \end{itemize}
  \end{itemize}
\end{itemize}
\end{document}

```

Результат компіляції:

- * Це перший рівень вкладення
- ** Це другий рівень вкладення
- *** Це третій рівень вкладення
- **** Це четвертий рівень вкладення

Створення нумерованого списку

Нумеровані списки формуються за допомогою оточення `enumerate`, яке дозволяє використовувати вкладені списки максимум до чотирьох рівнів вкладення. Нижче приведено приклад формування нумерованого списку із чотирма рівнями вкладення. Зверніть увагу на спосіб нумерації різних рівнів вкладення за замовчуванням:

```

\begin{enumerate}
  \item Це перший рівень вкладення
  \begin{enumerate}
    \item Це другий рівень вкладення
    \begin{enumerate}
      \item Це третій рівень вкладення
      \item Це третій рівень вкладення
      \item Це третій рівень вкладення
    \end{enumerate}
  \end{enumerate}
  \item Це четвертий рівень вкладення
\end{enumerate}

```

```

\end{enumerate}
\end{enumerate}
\end{enumerate}
  \item Це другий запис першого рівня
  \begin{enumerate}
    \item Це перший запис другого рівня
    вкладення
    \item Це другий запис другого рівня
    вкладення
    \item Це третій запис другого рівня
    вкладення
  \end{enumerate}
  \item Це третій запис першого рівня
\end{enumerate}

```

Результат компіляції:

1. Це перший рівень вкладення
 - (a) Це другий рівень вкладення
 - i. Це третій рівень вкладення
 - ii. Це третій рівень вкладення
 - iii. Це третій рівень вкладення
 - A. Це четвертий рівень вкладення
2. Це другий запис першого рівня
 - (a) Це перший запис другого рівня вкладення
 - (b) Це другий запис другого рівня вкладення
 - (c) Це третій запис другого рівня вкладення
3. Це третій запис першого рівня

Для нумерованих списків команда `\item` може мати необов'язковий параметр, який дозволяє задати нумерацію окремих елементів явним чином. Наприклад:

```

\begin{enumerate}
  \item[1] Це перший рівень вкладення
  \begin{enumerate}
    \item[1.1] Це другий рівень вкладення
  \end{enumerate}
\end{enumerate}

```

```

\item[2.1] Це третій рівень вкладення
\item[2.2] Це третій рівень вкладення
\item[2.3] Це третій рівень вкладення
\begin{enumerate}
\item[4.1] Це четвертий рівень вкладення
\end{enumerate}
\end{enumerate}
\end{enumerate}
\end{enumerate}

```

Результат компіляції:

- 1 Це перший рівень вкладення
 - 1.1 Це другий рівень вкладення
 - 2.1 Це третій рівень вкладення
 - 2.2 Це третій рівень вкладення
 - 2.3 Це третій рівень вкладення
 - 4.1 Це четвертий рівень вкладення

Пакет `enumerate`

Для формування нумерованих списків більш зручним та гнучким є пакет `enumerate`, який необхідно підключити в преамбулі документу: `\usepackage{enumerate}`. Пакет дозволяє задавати стиль нумерованого списку для кожного рівня вкладення самостійно за допомогою необов'язкового параметра оточення `enumerate`. Наприклад:

```

\begin{enumerate}[\#1]
\item Це перший рівень вкладення
\begin{enumerate}[№1]
\item Це другий рівень вкладення
\begin{enumerate}[Запис1]
\item Це третій рівень вкладення
\item Це третій рівень вкладення
\item Це третій рівень вкладення
\begin{enumerate}[Примітка1]
\item Це четвертий рівень вкладення
\end{enumerate}
\end{enumerate}
\end{enumerate}

```

```

\end{enumerate}
\item Це перший рівень вкладення
\end{enumerate}

```

Результат компіляції:

- #1 Це перший рівень вкладення
 - №1 Це другий рівень вкладення
 - Запис1 Це третій рівень вкладення
 - Запис2 Це третій рівень вкладення
 - Запис3 Це третій рівень вкладення
 - Примітка1 Це четвертий рівень вкладення
- #2 Це перший рівень вкладення

Створення списку літератури

Для створення списку літератури використовується оточення **`thebibliography`**. Порядок створення списку літератури зрозумілий із прикладу нижче:

```

\begin{thebibliography}{aaaaa}
%Література №1
\bibitem{lvov}\textit{Львовский С.М.}
Набор и верстка в пакете \LaTeX. М. : Космосинформ,
1995. -373с.
%Література №2
\bibitem{plis}\textit{Плис А. И., Сливина Н. А.}
MathCAD 2000:Математический практикум.
М.: финансы и статистика, 2002. 655 с.
%Література №3
\bibitem{polov}\textit{Половко Ф.М., Ганичев
И.В.} Mathcad для студента
СП.: БХВ-Петербург, 2006.- 226с.
%Література №4
\bibitem{spiv}\textit{Спивак М.} Восхитительный
TeX.М.: Мир, 1993. -285с
\end{thebibliography}

```

Результат компіляції:

Література

- [1] Львовский С.М. Набор и верстка в пакете \LaTeX . М. : Космосинформ, 1995. -373с.
- [2] Плис А. И., Сливина Н. А. MathCAD 2000:Математический практикум. М.: Финансы и статистика, 2002. 655 с.
- [3] Половко Ф.М., Ганичев И.В. Mathcad для студента СП.: ВХВ-Петербург, 2006.- 226с.
- [4] Спивак М. Восхитительный \TeX .М.: Мир, 1993. -285с

Заголовок до списку літератури генерується відповідно до стилю підключеного документу. При необхідності від може бути змінений в тілі документу командою (для документу зі стилем оформлення стаття):

```
\renewcommand {\refname}{Список першоджерел}
```

Даний приклад ілюструє зміну заголовку «Література» на «Список першоджерел». Аналогічно можна поміняти і інші стандартні заголовки в документі (див. Додаток №4).

Основними параметрами при використанні даного оточення є:

- {aaaaa}** –величина відступу списку літератури від лівої границі документу. В прикладі задано відступ 5 символів;
- \bibitem{lvov}** –починає кожен елемент списку літератури. Як обов'язковий параметр виступає мітка (деякий текст), що використовується для організації автоматичної генерації посилань на дане джерело;
- \textit** – обов'язковий параметр, що задає прізвище автора публікації.

Генерування посилань

Команда **\cite** дозволяє автоматично, за заданими мітками генерувати номери посилань на елементи списку літератури. Необов'язковий параметр команди може задавати додаткову інформацію стосовно згенерованого посилання. Наприклад:

```
%зміна заголовка до списку літератури
\renewcommand {\refname}{Список першоджерел}
Дане питання по роботі із видавничою системою
\LaTeX-детально розглянуто у \cite{ст.133}{lvov}
```

%генерація посилання із додатковою інформацією
та \cite{ст.112}{spiv}

```
Питання роботи із системою комп'ютерної
математики Mathcad розглядаються у \cite{plis}
%генерація посилання на літературу із міткою polov
та \cite{polov}
%формування списку літератури
\begin{thebibliography}{aaaaa}
\bibitem{lvov}\textit{Львовский С.М.}
Набор и верстка в пакете \LaTeX. М. :
Космосинформ, 1995. -373с.
\bibitem{plis}\textit{Плис А. И., Сливина Н. А.}
MathCAD 2000:Математический практикум.
М.: Финансы и статистика, 2002. 655 с.
\bibitem{polov}\textit{Половко Ф.М., Ганичев
И.В.} Mathcad для студента
СП.: ВХВ-Петербург, 2006.- 226с.
\bibitem{spiv}\textit{Спивак М.} Восхитительный TeX.
М.: Мир, 1993. -285с
\end{thebibliography}
```

Результат компіляції:

Дане питання по роботі із видавничою системою \LaTeX детально розглянуто у [1, ст.133] та [4, ст.112]

Питання роботи із системою комп'ютерної математики Mathcad розглядаються у [2] та [3]

Список першоджерел

- [1] Львовский С.М. Набор и верстка в пакете \LaTeX . М. : Космосинформ, 1995. -373с.
- [2] Плис А. И., Сливина Н. А. MathCAD 2000:Математический практикум. М.: Финансы и статистика, 2002. 655 с.
- [3] Половко Ф.М., Ганичев И.В. Mathcad для студента СП.: ВХВ-Петербург, 2006.- 226с.
- [4] Спивак М. Восхитительный \TeX . М.: Мир, 1993. -285с

Примітка. Для коректної генерації посилань на список літератури може знадобитися декілька компіляцій вихідного документу.

Зміст документу

LaTeX автоматично збирає інформацію для створення змісту документу і зберігає її в спеціальному файлі із ім'ям компільованого файлу та розширенням .toc. Для виводу змісту необхідно дати команду:

```
\tableofcontents
```

Зміст, що генерується, завжди відстає від реального стану справ, тому документ необхідно додатково скомпільовати принаймні ще один раз. Приклад формування змісту документу, що розглядався в роботі №2 (на ст.23) із дещо зміненими стильовими параметрами та додатковим розбиттям документу на окремі сторінки:

```
%перезначення заголовку
\renewcommand {\contentsname}{Зміст статті}
%формування змісту
\tableofcontents
```

Примітка. Перша команда ілюструє можливість зміни стандартного заголовку для змісту.

Результат компіляції:

Зміст статті

I	Все про LaTeX	2
1	Основи верстки документів	3
1.1	Стилі LaTeX	4
2	Робота з абзацами	6
2.1	Перенесення слів	7
2.1.1	Спосіб перенесення слів	8
2.1.2	Нерозривний пробіл	9

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скомпільуйте їх та перегляньте результат. При необхідності виправте помилки, що виникають при компіляції.
2. Створіть багатосторінковий документ (мінімум 4 сторінки): матеріал для документу – теоретичні відомості до роботи. В ньому використовуйте наступні елементи:
 - титульну сторінку із назвою статті та вашими даними: прізвищем та номером групи;
 - розмір шрифту документу – 12pt;
 - в титульному листі дату створення документу не виводіть;
 - матеріал документу почніть із нової сторінки;
 - створіть рубрики документа (розбити на глави, розділи, підрозділи і т.д.);
 - створіть список літератури (мінімум 4 джерела з посиланнями в тексті документу). Літературу візьміть із даного навчального посібника;
 - згенеруйте зміст документу;
 - поміняйте стандартні заголовки для списку літератури та змісту;
 - використовуйте марковані та нумеровані списки із кількома рівнями вкладення.

Контрольні питання

1. Яке оточення дозволяє створити маркований список?
2. Скільки рівнів вкладення може бути в маркованому списку?
3. Як поміняти вигляд маркера для одного маркованого списку?
4. Як поміняти вид маркерів для цілого документу?
5. Перерахуйте команди для створення нумерованого списку?
6. Яке призначення пакету *enumerate*?
7. Як створити список літератури?
8. Як поміняти заголовок до списку літератури?
9. Як в тексті документу вставити посилання на літературу?
10. Які додаткові параметри можна використати при генерації посилання на літературу можна використати?
11. Скільки компіляцій документу необхідно для остаточної генерації посилань на літературу?
12. Як згенерувати зміст документу?
13. Скільки компіляцій документу необхідно для остаточної генерації змісту документу?
14. Як поміняти заголовок до змісту документу?

Тема роботи: Робота із шрифтами. Ввід простих формул

Мета роботи: верстка документу у видавничій системі **LaTeX** з використанням гарнітур, розмірів шрифтів та простих формул

Для виконання роботи необхідно знати:

- загальну структуру документу **LaTeX**;
- команди управління гарнітурою шрифту;
- команди задання розміру шрифту;
- порядок вводу простих формул;
- використання в формулах основних математичних символів та роздільників;
- ввід тексту в математичні формули.

Теоретичні відомості

Робота із шрифтами

Для управління гарнітурою шрифту в **LaTeX** передбачено команди, приведені в таб. №12.

Таблиця №12. Команди управління гарнітурою шрифту

Команда	Альтернативна команда	Фрагмент документу	Результат компіляції
1	2	3	4
\bf	\textbf{}	\bf Напівжирний або \textbf{Напівжирний}	Напівжирний
\it	\textit{}	\it Курсив або \textit{Курсив}	Курсив
\sl	\textsl{}	\sl Похилий або \textsl{Похилий}	Похилий
\sf	\textsf{}	\sf Рублений або \textsf{Рублений}	Рублений
\sc	\textsc{}	\sc Капітель або \textsc{Капітель}	Капітель

1	2	3	4
<code>\tt</code>	<code>\texttt{}</code>	<code>\tt</code> Імітація др. машинки або <code>\texttt{Імітація др. машинки}</code>	Імітація др. машинки
<code>\rm</code>	<code>\textrm{}</code>	<code>\rm</code> Звичайний або <code>\textrm{Звичайний}</code>	Звичайний
<code>\em</code>	<code>\emph{}</code>	<code>\em</code> Виділений або <code>\emph{Виділений}</code>	Виділений

Команди першого стовпчика (можна використовувати одночасно кілька команд для створення комбінованої гарнітури) діють на весь текст, що слідує після них та, як правило, використовуються у групі, що задає область їх дії. Команди другого стовпчика мають обов'язків параметр – текст, на який вони діють. Наприклад:

`{\bf Принцип дії команди першого стовпчика.`

Її дія поширюється на весь текст,
що слідує після неї.

Для обмеження
області її дії { \it необхідно}
скористатися групою.}

Також можна `\textit{скористатися командами}`
другої групи.

Результат компіляції:

Принцип дії команди першого стовпчика.
Її дія поширюється на весь текст, що слідує після неї.
Для обмеження області її дії необхідно скористатися групою.
Також можна скористатися командами другої групи.

Розмір шрифту задається за допомогою команд, приведених у таб. №13. Для задання області дії команд необхідно використовувати групу.

Таблиця №13. Команди управління розміром шрифту

Команда	Назва шрифту	Результат компіляції
<code>\tiny</code>	крихітний	крихітний
<code>\scriptsize</code>	індексний	індексний
<code>\footnotesize</code>	Маленький	Маленький
<code>\small</code>	Менший від нормального	Менший від нормального
<code>\normalsize</code>	Нормальний	Нормальний
<code>\large</code>	Великий	Великий
<code>\Large</code>	Ще більший	Ще більший
<code>\LARGE</code>	Дуже великий	Дуже великий
<code>\huge</code>	Величезний	Величезний
<code>\Huge</code>	Велетенський	Велетенський

Розмір шрифту можна задати явним чином за допомогою команди `fontsize`:

`\fontsize{розмір шрифту}`
`{відстань між рядками }\selectfont`

тут:

розмір шрифту – розмір шрифту у будь-яких допустимих одиницях вимірювання;

відстань між рядками – міжрядковий інтервал, який, як правило, складає 110 -120% величини шрифту.

Наприклад:

`{\fontsize{24pt}{32pt}\selectfont}`
Приклад явного задання розміру шрифту та міжрядкового інтервалу.

`\fontsize{8pt}{10pt}\selectfont}`
Якщо команда задана за межами групи чи оточення, то вона буде стосуватися всього наступного тексту.

Розмір шрифту задається у будь-яких допустимих одиницях вимірювання.

`\large`

Відстань між рядками, як правило, не перевищує на 110-120~\% розмір шрифту).

Результат компіляції:

Приклад явного задання розміру шрифту та міжрядкового інтервалу.

Якщо команда задана за межами групи чи оточення, то вона буде стосуватися всього наступного тексту.

Розмір шрифту задається у будь-яких допустимих одиницях вимірювання.

Відстань між рядками, як правило, не перевищує на 110-120 % розмір шрифту).

Виділення тексту

Текст можна виділити за допомогою команд, описаних в таблиці №14. Можливості виділення тексту кольором будуть розглянуті в лабораторній роботі №10.

Таблиця №14. Команди для виділення тексту

Приклад команди	Спосіб виділення тексту та результат компіляції
1	2
риска <code>\overline{\tbox{зверху}}\text{}</code> тексту	Малює риску над фрагментом тексту: риска зверху тексту
приклад <code>\underline{підкреслення}</code> тексту	Малює риску під фрагменту тексту: приклад підкреслення тексту
це <code>\lowercase{Рядковий}</code> текст	Перетворює всі літери фрагменту тексту у рядкові: це рядковий текст
<code>\uppercase{Великі}</code> літери у тексті	Формує фрагмент тексту великими літерами: ВЕЛИКІ літери у тексті

1	2
текст <code>\raisebox{2pt}{вище}</code> на 2pt	Фрагмент тексту піднімається на задану величину над базовою лінією: текст вище на 2pt При від'ємному значенні параметру текст зсувається під базову лінію: текст вище на 2pt
текст <code>\fbox{обведений рамкою}</code>	Створює рамку навколо заданого фрагменту тексту. текст обведений рамкою
Створення <code>{\setlength{\fboxrule}{4pt}}</code> <code>}</code> <code>\setlength{\fboxsep}{6pt}</code> <code>\fbox{рамки товщиною 4pt з відступом від тексту 6pt}}</code>	Створює рамку навколо заданого фрагменту тексту. Параметр <code>\setlength{\fboxrule}{4pt}</code> задає товщину рамки 4pt. Параметр <code>\setlength{\fboxsep}{4pt}</code> визначає проміжок між рамкою та текстом 4pt. Створення рамки товщиною 4pt з відступом від тексту 6pt

Ввід простих формул

В документі можна використовувати формули двох типів:

- **формули, що входять у текст:** розміщуються між обмежувачами `$` або `\(i \)`. Наприклад:

Обчислити значення функції

$$f(x) = ax^2 + bx + c \text{ при } x = 3.24$$

Результат компіляції:

Обчислити значення функції $f(x) = ax^2 + bx + c$ при $x = 3.24$.

- **«виключні формули»:** розміщуються за межами тексту у окремому рядку. Створюються за допомогою подвійних обмежувачів `$$`, або `\[i \]`, або з використанням оточення `equation`. Наприклад:

Обчислити значення функції

$$f(x) = ax^2 + bx + c \text{ при } x = 3.24$$

Результат компіляції:

Обчислити значення функції

$$f(x) = ax^2 + bx + c$$

при

$$x = 3.24$$

Основні правила верстки формул:

- формула утворює групу;
- кожна буква у формулі розглядається як ім'я змінної та друкується математичним курсивом;
- формулами можуть бути і окремі символи;
- пропуски в середині формул ігноруються;
- порожні рядки в середині формул є недопустимими;
- для «виключної формули», та формули в тексті документу результати компіляції дещо відрізняються як за розміром так і способом розстановки надрядкових та підрядкових символів.

Спеціальні математичні символи можуть бути введені як команди безпосередньо із клавіатури (див. додаток №5), або за допомогою відповідної палітри редактора **Kile (WinEdt)** (див. додаток №2 або №3).

Формули, що входять в текст, при компіляції можуть бути розірвані на два рядки. Для виключення такої ситуації необхідно скористатися фігурними дужками: формула має міститися в оточенні:

$\$(формула)\$$.

Виключна формула завжди формується тільки в межах одного рядка, і при цьому вона може вийти за допустимі межі вводу тексту. Для виключення такої ситуації необхідності формувати формулу у декілька рядків, штучно розбивши її на окремі фрагменти, які оформити незалежно один від одного: більш детально ці питання будуть розглянуті у наступній роботі.

Ввід простих формул: відступи тексту від формули

Відступи тексту від формули можуть бути змінені за допомогою команди `\setlength`. Так, якщо задати приведену нижче команду, то формула в тексті буде додатково відділена з обох сторін відступами `3pt`:

`\setlength{\mathsurround}{3pt}`

Зверніть увагу на різне оформлення формул в тексті у прикладі нижче:

`{\bfТеорема Піфагора}: У прямокутному трикутнику квадрат гіпотенузи $\$c\$$ рівний сумі квадратів катетів $\$a\$$ та $\$b\$$: $\$c^2=a^2+b^2\$$.`

`%задання додаткового відступу тексту від формул
\setlength{\mathsurround}{3pt}`

`{\bfТеорема Піфагора}: У прямокутному трикутнику квадрат гіпотенузи $\$c\$$ рівний сумі квадратів катетів $\$a\$$ та $\$b\$$: $\$c^2=a^2+b^2\$$.`

Результат компіляції:

Теорема Піфагора: У прямокутному трикутнику квадрат гіпотенузи c рівний сумі квадратів катетів a та b : $c^2 = a^2 + b^2$.

Теорема Піфагора: У прямокутному трикутнику квадрат гіпотенузи c рівний сумі квадратів катетів a та b : $c^2 = a^2 + b^2$.

Степені та індекси

Степені та індекси вводяться за допомогою спеціальних символів $^$ та $_$ відповідно. Якщо індекс чи показник степеня складається із кількох символів, то необхідно створити групу – використати фігурні дужки. Порядок задання верхнього та нижнього індексів для однієї змінної значення не має. Наприклад:

$\$ \$$

$$y=a_1x^2+b_1x+c_{22}^3$$

$\$ \$$

Результат компіляції:

$$y = a_1x^2 + b_1x + c_{22}^3$$

Індекси у формулах, що знаходяться в межах тексту, та у «виключних» формулах відображаються по різному. Наприклад:

Це є приклад створення "виключної" формули. По замовчуванню вона знаходиться по центру рядка, а індекси знаходяться над відповідними математичними символами

$\$ \$$

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Ця ж формула в межах тексту матиме наступний вигляд:

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Результат компіляції:

Це є приклад створення "виключної" формули. По замовчуванню вона знаходиться по центру рядка, а індекси знаходяться над відповідними математичними символами

$$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$$

Ця ж формула в межах тексту матиме наступний вигляд: $\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Розміщенням індексів у формулах можна управляти за допомогою команд `\limits` та `\nolimits`. Результат їхньої дії зрозумілий із приведеного нижче прикладу:

"Виключна формула"

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Формула в межах тексту:

$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Результат компіляції:

"Виключна формула"

$$\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$$

Формула в межах тексту: $\int_{x_1}^{x_2} f(x) dx = h \sum_{i=0}^{ny_i}$

Дроби

Дроби, що позначаються похилою рисою, вводяться безпосередньо за допомогою символу `/`. Наприклад:

$y = a_1/b_1^2$

Результат компіляції

$$y = a_1/b_1^2$$

Дроби, в яких чисельник та знаменник розділені горизонтальною рисою, створюються командою `\frac` із двома обов'язковими аргументами – чисельником та знаменником.

Приклад №1:

Формула $y = \frac{a_1}{b_1^2}$ знаходиться в тексті

Результат компіляції:

Формула $y = \frac{a_1}{b_1^2}$ знаходиться в тексті

Приклад №2:

Формула $y = \frac{a_1}{b_1^2}$ знаходиться за межами тексту: є "виключною"

Результат компіляції:

Формула

$$y = \frac{a_1}{b_1^2}$$

знаходиться за межами тексту: є "виключною"

Примітка: у формулах, що входять у текст документу, чисельник та знаменник друкуються розміром шрифту, дещо меншим ніж сам текст документу.

Штрихи та похідні

Штрихи та похідні вводяться за допомогою одинарних лапок. Наприклад:

Перша похідна функції $f(x) = ax^2 + bx + c$ рівна $f'(x) = 2ax + b$, а друга – $f''(x) = 2a$.

Результат компіляції:

Перша похідна функції $f(x) = ax^2 + bx + c$ рівна $f'(x) = 2x + b$, а друга – $f''(x) = 2$.

Корені

Корені вводяться командою `\sqrt` із обов'язковим параметром – підкореневим виразом та необов'язковим – степенем кореня. Наприклад:

```
$$y=\frac{\sqrt{a_1}}{\sqrt[3]{b_1^2}}$$
```

Результат компіляції:

$$y = \frac{\sqrt{a_1}}{\sqrt[3]{b_1^2}}$$

Трикратка

Вводиться за допомогою команди `\dots` (а не вводом відповідної кількості крапок. Наприклад:

```
$i=1,3,\dots,(2n+1)$
```

Результат компіляції:

$$i = 1, 3, \dots, (2n + 1)$$

Перекреслені символи

Для набору перекресленого символу необхідно перед командою, що задає даний символ, вставити команду `\not`. Наприклад:

```
$a \not \in A$
```

Результат компіляції:

$$a \notin A$$

Примітка: деякі символ будуть перекреслено некоректно, тому для їх вводу необхідно скористатися вставкою відповідного перекресленого символу.

Дужки

Прості та квадратні дужки вводяться із клавіатури безпосередньо, а фігурні – за допомогою команд `\{` та `\}` відповідно. Для введення дужок, розмір яких по висоті перевищує висоту стандартного символу, служать команди: `\left` для лівої дужки та `\right` для правої (такі дужки по висоті будуть відповідати фрагменту формули, що міститься між ними. Команди можуть використовуватися для всіх типів дужок (див. додаток

№5: Роздільники). Кожній команді `\left` повинна відповідати команда `\right`, бо інакше при компіляції буде видано повідомлення про помилку. Приклад використання дужок різного розміру:

```
$$y=(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}})$$
$$y=\left(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}}\right)$$
```

Результат компіляції:

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

Їх розмір можна задати і явно за допомогою команд (у порядку зростання їх розміру) `\bigl`, `\Bigl`, `\biggl` і `\Biggl` для лівих та відповідно `\bigr`, `\Bigr`, `\biggr` і `\Biggr` для правих дужок. Наприклад:

```
$$y=\bigl(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}}\bigr)$$
$$y=\Bigl(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}}\Bigr)$$
$$y=\biggl(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}}\biggr)$$
$$y=\Biggl(\frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}}\Biggr)$$
```

Результат компіляції:

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right)$$

Якщо одну із дужок відображати не потрібно, то замість неї просто ставиться крапка. Наприклад:

```
% Формула знаходиться в межах документа

$$\left. \frac{ax+b}{d} \right|_a^b$$

% «виключна» формула

$$\left. \frac{ax+b}{d} \right|_a^b$$

```

Результат компіляції:

$$\left. \frac{ax+b}{d} \right|_a^b$$

$$\left. \frac{ax+b}{d} \right|_a^b$$

Текст в формулах

Безпосередньо в формули текст вводити не можна. Для цього необхідно скористатися командою `\mbox` або `\text`. Наприклад:

```
Даний фрагмент ілюструє можливість
вставки тексту у формулу:

$$f(x) = ax^2 + bx + c \text{ при } x = 2 \text{ рівна } 0$$

```

Результат компіляції:

Даний фрагмент ілюструє можливість вставки тексту у формулу:

$$f(x) = ax^2 + bx + c \text{ при } x = 2 \text{ рівна } 0$$

Зверніть увагу, що в `\mbox` для додаткового відступу тексту від формул вставлено додаткові пробіли.

Математичні функції

Імена, прийняті для позначення математичних функцій *sin*, *cos*, *tg*,... друкуються командами, складеними з `\` та імені функції. Між ім'ям функції та аргументом автоматично вставляється невеликий відступ. Для вводу аргументу функції в дужки їх треба набирати самому. Перелік команд для друку імен математичних функцій подано в додатку №5: Команди для друку імен математичних функцій. Наприклад:

Правильно ввід імені функції `\sin(\omega t + \varphi)`

Неправильно ввід імені функції `\sin(\omega t + \varphi)`

Результат компіляції:

Правильно ввід імені функції $\sin(\omega t + \varphi)$
Неправильно ввід імені функції $\sin(\omega t + \varphi)$

Розміщення одного символу над іншим

Команда `\atop` друкує одну частину формули трохи нижче рядка, а другу частину трохи вище: при цьому символи знаходяться один над одним. Наприклад:

`\mathbf{x+y} \atop q_i`

Результат компіляції:

$$\mathbf{x+y} \atop q_i$$

Команда `\choose` аналогічна до попередньої команди, тільки вираз виводиться у дужках:

$$\binom{x+y}{q_i}$$

Форматування тексту у математичну формулу

Для форматування тексту у математичній формулі можуть бути використані наступні команди (звичайні команди форматування тексту можна також використовувати у межах команд `\mbox` або `\text`):

`\mathrm{}` – прямий шрифт;
`\mathbf{}` – напівжирний шрифт;
`\mathit{}` – курсивний шрифт;
`\mathsf{}` – «рублений» шрифт;
`\mathtt{}` – машинописний шрифт.

Команди працюють тільки з текстом, набраного англійськими буквами. Наприклад:

```
$$ \mathrm{ \mathrm{ } } $$
$$ \mathbf{ \mathbf{ } } $$
$$ \mathit{ \mathit{ } } $$
$$ \mathsf{ \mathsf{ } } $$
$$ \mathtt{ \mathtt{ } } $$
```


Результат компіляції:

mathrm

mathbf

mathit

mathsf

mathtt

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте їх та перегляньте результат. При необхідності виправте помилки.
2. Створіть багатосторінковий документ: матеріал для документу – теоретичні відомості до даної роботи. Використайте команди для роботи із гарнітурою шрифту, його розміром та можливістю виділення фрагментів тексту.
3. Відповідно до індивідуального завдання наберіть на окремій сторінці документу формули в тексті, а на другій зробіть їх виключними.

Завдання для індивідуального виконання

№ варіанту	Завдання	№ варіанту	Завдання
1	2	3	4
1.	$x^2 + y^3 = 2x + c$ $12 + \frac{19}{3} + \frac{26}{5} + \dots + \frac{5+7n}{2n-1}$ $\frac{\sqrt[3]{8}}{5} - \frac{\sqrt[3]{9}}{25} + \frac{\sqrt[3]{10}}{125} - \dots + (-1)^{n+1} \frac{\sqrt[3]{n+7}}{5^n}$ $0.123^2 + \frac{32.265 + 2.56^3}{\sqrt[3]{456 + 3.12^{-0.236}}}$	16.	$z = x^3 + 8y^3 - 6xy + 5$ $\frac{1}{1 \cdot 4} + \frac{1}{4 \cdot 7} + \frac{1}{7 \cdot 10}$ $\frac{\sqrt[3]{10}}{6} - \frac{\sqrt[3]{11}}{36} + \frac{\sqrt[3]{12}}{216} - \dots + (-1)^{n+1} \frac{\sqrt[3]{n+9}}{6^n}$ $\frac{45.23 + 2.65^3}{5.26} - \sqrt{23.56^2 + 3.26^4}$
2.	$(x^2 + y^2) = 2ax^3$ $\frac{6}{5} + \frac{11}{23} + \frac{16}{53} + \dots + \frac{5n+1}{6n^2-1}$ $-\frac{1}{\sqrt{5}} + \frac{1}{\sqrt{8}} - \frac{1}{\sqrt{11}} + \dots + (-1)^n \frac{1}{\sqrt{3n+2}}$ $\frac{23.6^2 + 3.25^3}{2.654} + \sqrt[3]{2.65 + 3.25^4}$	17.	$z = 1 + 6x - x^2 - xy - y^2$ $\frac{3}{8} + \frac{5}{23} + \frac{7}{48} + \dots + \frac{2n+1}{5n^2+3}$ $z = \frac{\sqrt[3]{xy - x^2 + 1}}{2x - 3y}$ $\sqrt[6]{12.26 + 2.65^3 + 5.26^{-1.26}}$

1	2	3	4
3.	$(x^2 + y^2)^3 = 4x^2y^2$ $1 - \frac{6}{9} + \frac{11}{27} - \dots + (-1)^{n+1} \frac{n^2 + 2}{3^n}$ $xy^3 + yx^3 + \sqrt{1 + (xy^3 + yx^3)^2}$ $\sqrt{\frac{2.65^3 + 0.265^2}{15.26}} + 0.12^4$	18.	$z = x^3 + y^2 - 6xy - 39x + 18y + 20$ $1 - \frac{1}{2^2} + \frac{1}{3^3} - \frac{1}{4^4}$ $-\frac{1}{\sqrt[3]{3}} + \frac{1}{\sqrt[3]{4}} - \frac{1}{\sqrt[3]{5}} + \dots + (-1)^n \frac{1}{\sqrt[3]{n+2}}$ $15 \cdot 26^3 + \sqrt[6]{\frac{12 \cdot 23^4 + 3 \cdot 26^5}{156 \cdot 23 + 2 \cdot 26}}$
4.	$x^2 + y^2 = R^2$ $\frac{1}{1 \cdot 2^2} + \frac{2}{3 \cdot 2^3} + \frac{3}{5 \cdot 2^4}$ $-\frac{1}{\sqrt{6}} + \frac{1}{3} - \frac{1}{\sqrt{14}} + \dots + (-1)^n \frac{1}{\sqrt{n^2 + 5}}$ $45 \cdot 26^2 + 5.65 + \sqrt{\frac{2.65^3 + 0.265^2}{15.26}}$	19.	$z = 4(x - y) - x^2 - y^2$ $\frac{2}{8} - \frac{5}{64} + \frac{10}{512} - \dots + (-1)^{n+1} \frac{n^2 + 1}{8^n}$ $\sqrt{5} + \sqrt{\frac{13}{3}} + \sqrt{\frac{21}{5}} + \dots + \sqrt{\frac{8n-3}{2n-1}}$ $y(a, \theta) = \left(\frac{2 \cdot a^2 + b^2}{3 \cdot a + b} \right)^{\frac{3}{2}}$
5.	$z^2 = x^2 + y^2$ $6 + \frac{27}{15} + \frac{62}{80} + \dots + \frac{7n^2 - 1}{2n^3 - 1}$ $-\frac{1}{\sqrt[3]{5}} + \frac{1}{\sqrt[3]{6}} - \frac{1}{\sqrt[3]{7}} + \dots + (-1)^n \frac{1}{\sqrt[3]{n+4}}$ $s(a, x, e) = \left(\frac{x + 3 \cdot a + y}{2 \cdot x} \right)^4$	20.	$z = 4xy - 2x^2 - y^2 + 5$ $y = \frac{x^2 + 3xy - y^2}{3x^2 - 2xy}$ $u = 2\sqrt{\frac{1 - \sqrt{xyz}}{1 + \sqrt{xyz}}}$ $\frac{12 \cdot 26 + 3 \cdot 26^3}{12 \cdot 26} - 2.654 \cdot \sqrt[3]{2 \cdot 26 + 1 \cdot 26^3}$
6.	$x^2 + y^2 = 2x$ $2 + \frac{5}{10} + \frac{10}{25} + \dots + \frac{n^2 + 1}{3n^2 - 2}$ $\frac{\sqrt[3]{9}}{8} - \frac{\sqrt[3]{10}}{64} + \frac{\sqrt[3]{11}}{512} - \dots + (-1)^{n+1} \frac{\sqrt[3]{n+8}}{8^n}$ $w(x, y, z) = \cos \left[\sqrt[3]{\frac{(x + y + z)^2}{(x \cdot y \cdot z)^{\frac{2}{3}}}} \right]$	21.	$z = x^2 + 2xy - y^2 - 2x + 2y$ $3y = \frac{y^2}{x^2} + 8 \frac{y}{x} + 4$ $\frac{1}{\sqrt{5}} + \frac{\sqrt{2}}{\sqrt{9}} + \frac{\sqrt{3}}{\sqrt{13}} + \dots + \sqrt{\frac{n}{4n+1}}$ $\sqrt[3]{12 \cdot 26^2 + 3 \cdot 26^3} + 6 \cdot 26^{-0.265} - \sqrt{456 \cdot 26}$
7.	$x^2 + y^2 + z^2 = R^2$ $\frac{2}{7} - \frac{6}{49} + \frac{12}{343} - \dots + (-1)^{n+1} \frac{n(n+1)}{7^n}$ $u = 2\sqrt{\frac{1 - \sqrt{xyz}}{1 + \sqrt{xyz}}}$ $2.65^{-0.32} \cdot 3.26^2 + 0.26 \cdot \sqrt[3]{12 \cdot 26 + 2 \cdot 65^3}$	22.	$z = 2x^3 + 2y^3 - 6xy + 5$ $xy' = \frac{3y^3 + 6yx^2}{2y^2 + 3x^2}$ $\frac{1}{\sqrt{1 \cdot 2}} + \frac{1}{\sqrt{2 \cdot 3}} + \frac{1}{\sqrt{3 \cdot 4}}$ $\sqrt{2 \cdot 26^2 + 3 \cdot 26^3} + 0.5 \cdot 2 \cdot 26^4 + \frac{126}{1 \cdot 2 - 0.456}$

1	2	3	4
8.	$(x^2 + y^2)^3 = 9x^2(4x^2 + 3y^2)$ $6 + \frac{7}{9} + \frac{8}{25} + \dots + \frac{n+5}{(2n-1)^2}$ $-\frac{1}{3} + \frac{1}{\sqrt{12}} - \frac{1}{\sqrt{17}} + \dots + (-1)^n \frac{1}{\sqrt{n^2+8}}$ $0.25 \cdot \sqrt[3]{12.26} + 3.26^4 + 5.26^{-1.26}$	23.	$z = x^2 + xy + y^2 + x - y + 1$ $xy = \frac{3y^3 + 8yx^2}{2y^2 + 4x^2}$ $u = 2 \sqrt[3]{\frac{1 - \sqrt{xyz}}{1 + \sqrt{xyz}}}$ $\frac{12.26 + 3.26^3}{12.26} - 2.654 \cdot \sqrt[3]{2.26 + 1.26^3}$
9.	$x^4 = 16(3x^2 - y^2)$ $\frac{5}{6} + \frac{7}{13} + \frac{9}{20} + \dots + \frac{3+2n}{7n-1}$ $3 + \frac{4}{\sqrt{28}} + \frac{5}{\sqrt{73}} + \dots + \frac{n+2}{\sqrt{9n^2-8}}$ $w(x, y, b) = \frac{b^2 \cdot x - x^3 + b}{b \cdot y^{x+3}}$	24.	$z = (x-5)^2 + y^2 + 1$ $y' = \frac{x^2 + 3xy - y^2}{3x^2 - 2xy}$ $\frac{\sqrt[3]{9}}{8} - \frac{\sqrt[3]{10}}{64} + \frac{\sqrt[3]{11}}{512} - \dots + (-1)^{n+1} \frac{\sqrt[3]{n+8}}{8^n}$ $\sqrt[3]{12.26^2 + 3.26^3} + 6.26^{-0.254}$
10.	$y^6 = 25(y^4 - x^4)$ $\frac{5}{4} + \frac{7}{9} + \frac{9}{16} + \dots + \frac{2n+3}{(n+1)^2}$ $\frac{1}{\sqrt{5}} + \frac{\sqrt{2}}{\sqrt{9}} + \frac{\sqrt{3}}{\sqrt{13}} + \dots + \frac{\sqrt{n}}{\sqrt{4n+1}}$ $e^{-ax} \cdot \sqrt{x+1} + e^{-bx} \cdot \sqrt[3]{x+1.5}$	25.	$z = 2x^2y - x^3y - x^2y^2$ $xy = \frac{3y^3 + 12yx^2}{2y^2 + 6x^2}$ $-\frac{1}{\sqrt{5}} + \frac{1}{\sqrt{8}} - \frac{1}{\sqrt{11}} + \dots + (-1)^n \frac{1}{\sqrt{3n+2}}$ $\frac{x + \arccos 2x}{x + \sqrt{a+b} \sin 3c}$
11.	$(x^2 + y^2)^2 = 4(2x^2 + 3y^2)$ $5 - \frac{13}{4} + \frac{18}{8} - \dots + (-1)^{n+1} \frac{n^2+9}{2^n}$ $z = \frac{\sqrt[3]{xy - x^2 + 1}}{2x - 3y}$ $e^{-ax} \cdot \frac{x + \sqrt{x+a}}{c - \sqrt{2x+b}}$	26.	$z = 3x^3 + 3y^3 - 9xy + 10$ $4y' = \frac{y^2}{x^2} + 10 \frac{y}{x} + 5$ $\sqrt{5 + y^2} dx + 4(x^2y + y) dy = 0$ $\frac{a}{x+2} e^{-bx^2} + \ln(a+bx)$
12.	$y^6 = (x^2 + y^2)(3y^2 - x^2)$ $-\frac{1}{5} + \frac{1}{13} - \frac{1}{21} + \dots + (-1)^n \frac{1}{8n-3}$ $\frac{1}{\sqrt{1 \cdot 2}} + \frac{1}{\sqrt{2 \cdot 3}} + \frac{1}{\sqrt{3 \cdot 4}}$ $k(x, y) = \cos\left(\frac{x-y}{x+y}\right) + e^{\frac{y}{x}}$	27.	$z^3 + 3x^2z = 2xy$ $xy = \frac{3y^3 + 14yx^2}{2y^2 + 7x^2}$ $x\sqrt{4+y^2} dx + y\sqrt{1+x^2} dy = 0$ $\frac{ax + e^{-x} \cos bx}{bx - e^{-x} \sin bx + 1}$

1	2	3	4
13.	$z = 2x^{-y} + y^2(x + y^2)$ $\frac{3}{4} - \frac{8}{16} + \frac{15}{64} - \dots + (-1)^{n+1} \frac{n(n+2)}{4^n}$ $(-1)^n \frac{1}{\sqrt[3]{n+5}}$ $w(x, y, b) = \frac{b^2 \cdot x - x^3 + b}{b \cdot y^{x+3}}$	28.	$z = 6(x-y) - 3x^2 - 3y^2$ $3y = \frac{y^2}{x^2} + 10 \frac{y}{x} + 10$ $x\sqrt{5+y^2} dx + y\sqrt{4+x^2} dy = 0$ $y(x) = \cos\left(\frac{x}{x+1.7}\right)^2 + \frac{x}{x^2-2.5}$
14.	$z = 2x^2 + 2xy - \frac{1}{2}y^2 - 4x$ $\frac{4}{9} + \frac{7}{49} + \frac{10}{121} + \dots + \frac{3n+1}{(4n-1)^2}$ $u = 2 \sqrt[3]{\frac{1 - \sqrt{xyz}}{1 + \sqrt{xyz}}}$ $y(x) = \frac{x}{x^2+1} - \sqrt{ \sin(x) }$	29.	$z = x^2 + xy + y^2 - 6x - 9y$ $y = \frac{x^2 + 2xy - 5y^2}{2x^2 - 6xy}$ $x\sqrt{3+y^2} dx + y\sqrt{2+x^2} dy = 0$ $k(x, y) = \cos\left(\frac{x-y}{x+y}\right) + e^{\frac{y}{x}}$
15.	$z = 6xy - 9x^2 - 9y^2 + 4x + 4y$ $\frac{1}{8} + \frac{10}{15} + \frac{19}{22} + \dots + \frac{9n-8}{1+7n}$ $\frac{\sqrt{2}}{1 \cdot 2} + \frac{\sqrt{3}}{2 \cdot 3} + \frac{\sqrt{4}}{3 \cdot 4}$ $y(x) = \sin\left(\frac{1}{x^2} + \frac{1}{x}\right)^2 + \ln\left(\frac{x}{5}\right)^{\frac{3}{2}}$	30.	$z = x^2 - 2y^2 + 4xy - 6x - 1$ $2y = \frac{y^2}{x^2} + 8 \frac{y}{x} + 8$ $y'y \sqrt{1-x^2} + 1 = 0$ $y(x) = \frac{\ln(x^2)}{\sin(x-0.5)} + \cos(\sqrt{ x })$

Контрольні питання

1. Перерахуйте основні типи гарнітур шрифту.
2. Як поміняти гарнітуру шрифту?
3. Як задати розмір шрифту для всього документу?
4. Як поміняти розміру шрифту в межах групи?
5. Чи можна одночасно застосувати кілька гарнітур шрифту?
6. Як провести лінію над фрагментом тексту?
7. Як підняти текст над базовою лінією?
8. Як виділити фрагмент тексту за допомогою рамки?
9. Як поміняти параметри рамки для виділення тексту?
10. Які типи формул ви знаєте?
11. Яка формула називається «виключною»?
12. Перерахуйте особливості верстки формул.
13. Як в формулу вставити індекси?
14. Які типи дробових рисок можна використати в формулах та як їх вставити?
15. Перерахуйте основні типи дужок, що можуть використовуватися в формулах?
16. Для чого використовуються команди `\left` і `\right`?
17. Як вставити текст в формулу?
18. Як задати додаткову відстань від формули до тексту, що в неї вставлений?

Лабораторна робота №6

Тема роботи: Ввід складних формул у видавничій системі LaTeX

Мета роботи: верстка документу з використанням додаткових можливостей LaTeX для створення складних формул

Для виконання роботи необхідно знати:

- оточення `equation` та його можливості;
- використання оточення `eqnarray` для створення багаторядкових формул;
- спеціалізовані оточення пакету `amsmath`.

Теоретичні відомості

Оточення `equation`

Оточення `equation` дає можливість оформити математичні формули із використанням нумерації та автоматичної генерації посилань. Виключна формула, сформована за допомогою оточення, автоматично отримує номер. Для посилання на формулу із будь-якого місця документу їй присвоюється мітка командою `\label` (обов'язковий параметр команди – мітка формули):

`\label{мітка}`

Посилання на формулу в тексті можна задати за допомогою однієї із команд (як обов'язковий параметр виступає мітка формули, на яку виконується посилання):

- `\ref` генерування посилання тільки на номер формули;
- `\pageref` генерування посилання і на номер сторінки, на яку попадає формула.

Наприклад:

`\ref{мітка}`

При необхідності самостійно нумерувати формули можна скористатися командами `\eqno` (номер справа від формули) та `\leqno` (номер зліва від формули): автоматичне генерування посилань при цьому не передбачається. Приклад використання оточення для створення виключних формул, організації їх нумерації та створення посилань на формулу у тексті документу:

```

begin{equation}
%мітка першої формули
\label{Formula1}
% перша формули
y=\bigl( \frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}} \bigr)
\end{equation}
\begin{equation}
%мітка другої формули
\label{Formula2}
% друга формули
y=\Bigl( \frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}} \Bigr)
\end{equation}
\begin{equation}
%мітка третьої формули
\label{Formula3}
% третя формула
y=\biggl( \frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}} \biggr)
\end{equation}
\begin{equation}
%мітка четвертої формули
\label{Formula4}
% четверта формули
y=\Biggl( \frac{\sqrt{a_1}+1}{\sqrt[3]{b_1^2}} \Biggr)
\end{equation}
% нумерація формул вручну
$$ 2^2=4\eqno (\textcircled{e}) $$
$$ 2^2=4 \leqno (*) $$

```

%посилання в тексті на першу формулу
Перша формула на сторінці `\pageref{Formula1}` ілюструє вставку дужок за допомогою команд `bigl--bigr`.

%посилання в тексті на другу формулу
Формула номер `(\ref{Formula2})` ілюструє вставку дужок за допомогою команд `Bigl--Bigr`.

%посилання в тексті на четверту формулу
Формула `(\ref{Formula2})` на сторінці `\pageref{Formula4}` ілюструє вставку дужок за допомогою команд `Biggl--Biggr`.

Результат компіляції:

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right) \quad (4)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right) \quad (5)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right) \quad (6)$$

$$y = \left(\frac{\sqrt{a_1} + 1}{\sqrt[3]{b_1^2}} \right) \quad (7)$$

$$2^2 = 4 \quad (\textcircled{e})$$

$$(*) \quad 2^2 = 4$$

Перша формула на сторінці 3 ілюструє вставку дужок за допомогою команд `bigl--bigr`. Формула номер (9) ілюструє вставку дужок за допомогою команд `Bigl--Bigr`.

Формула (9 на сторінці 3) ілюструє вставку дужок за допомогою команд `Biggl--Biggr`.

Системи рівнянь

Системи рівнянь в **LaTeX**, як правило, створюють за допомогою оточення `eqnarray` для нумерованих та `eqnarray*` для нумерованих формул. В оточенні рівняння, що розміщуються на окремих рядках, відділяються одне від одного командами `\\`. Вираз в межах рядка складається з трьох частин (можна порожніх), розділених `&`. Кожна частина розміщується в своєму стовпі. В лівому стовпці формули вирівнюються по правому краю, в середньому – по центру, а в правому – по лівому краю. Символи `&` задають точки вирівнювання формул різних рядків по вертикалі. Кожен рядок в оточенні `eqnarray` нумерується. Відмінити нумерацію будь-якого рядка можна командою `\nonumber`. Приклад фрагменту документа для набору кількох формул подано нижче:

```

\begin{eqnarray}
%перша формула
C=2 \pi r \& \approx 6.283r \quad ||
%друга формула
S=\pi r_2 \& \approx 3.142r^2 \quad ||

```

```
%виключення нумерації третьої формули
\nonumber
%третя формула
d=2 \sqrt{s/\pi} &\approx 1.28 \sqrt{S} \\
%четверта формула
\pi= \frac{C}{d} &\approx 3.141592653589792 \ldots
\end{eqnarray}
```

Результат компіляції:

$$\begin{aligned} C &= 2\pi r \approx 6.283r & (1) \\ S &= \pi r_2 \approx 3.142r^2 & (2) \\ d &= 2\sqrt{s/\pi} \approx 1.28\sqrt{S} \\ \pi &= \frac{C}{d} \approx 3.141592653589792 \dots & (3) \end{aligned}$$

Оточення `equation` і `eqnarray` використовують один лічильник `equation`, значення якого друкується як номер формули, тому всі нумеровані формули матимуть єдину нумерацію.

Системи рівнянь без вирівнювання

Для системи рівнянь без вирівнювання по вертикалі в пакеті `amsmath` визначено оточення `gather`. Воно дозволяє створювати нумеровані системи рівнянь (ненумеровані створює оточення `gather*`). Місця перенесення рядків задаються командою `\\`, а всі формули в рядках розміщуються по центру. Приклад використання оточення:

```
begin{gather}
%перша формула
C=2 \pi r \approx 6.283r \\
%друга формула
\nonumber
S=\pi r_2 \approx 3.142r^2 \\
\nonumber
%третя формула
d=2 \sqrt{s/\pi} \approx 1.28 \sqrt{S} \\
%четверта формула
\nonumber
\pi= \frac{C}{d} \approx 3.141592653589792 \ldots
\end{gather}
```

Результат компіляції:

$$\begin{aligned} C &= 2\pi r \approx 6.283r & (1) \\ S &= \pi r_2 \approx 3.142r^2 \\ d &= 2\sqrt{s/\pi} \approx 1.28\sqrt{S} \\ \pi &= \frac{C}{d} \approx 3.141592653589792 \dots \end{aligned}$$

Системи рівнянь із вирівнюванням

Для систем рівнянь із додатковим вирівнюванням по вертикалі в пакеті `amsmath` визначено оточення `align` для нумерованих, та `align*` для ненумерованих формул. Точки вирівнювання рівнянь задаються `&`. На відміну від `eqnarray` немає необхідності штучно розбивати рівняння на три частини:

```
\begin{align}
%перша формула
C=& 2 \pi r \approx 6.283r \\
%друга формула
\nonumber
S=& \pi r_2 \approx 3.142r^2 \\
\nonumber
%третя формула
d=& 2 \sqrt{s/\pi} \approx 1.28 \sqrt{S} \\
%четверта формула
\nonumber
\pi=& \frac{C}{d} \approx 3.141592653589792 \ldots
\end{align}
```

Результат компіляції:

$$\begin{aligned} C &= 2\pi r \approx 6.283r & (1) \\ S &= \pi r_2 \approx 3.142r^2 \\ d &= 2\sqrt{s/\pi} \approx 1.28\sqrt{S} \\ \pi &= \frac{C}{d} \approx 3.141592653589792 \dots \end{aligned}$$

Розбиття довгих формул

Виключні формули `LaTeX` не переносить. Якщо така формула не поміщається в рядку, то при трансляції видається повідомлення про переповнення (`Overfull ...`). Для вводу довгих формул їх доводиться розбивати

[illegible]
$$\begin{aligned} & \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}s\right) dt \\ & \approx \frac{b-a}{2} \sum_{j=1}^N c_j^{(n)} f\left(\frac{a+b}{2} + \frac{b-a}{2}s_j\right) \end{aligned}$$

```

\begin{eqnarray}
\nonumber
%перший фрагмент формули
\int_{t_j}^{t_{j+1}} K(x,s)
\varphi(s)ds =& \& \backslash
%другий фрагмент формули
\frac{h}{2}\int_{-1}^1 \&
K(x,\frac{t_{j+1}+t_j}{2}) +
\frac{h}{2}s) \varphi(\frac{t_{j+1}+t_j}{2}) +
\frac{h}{2}s)ds \&
\end{eqnarray}

```

$$\int_{t_j}^{t_{j+1}} K(x, s) \varphi(s) ds = \frac{h}{2} \int_{-1}^1 K(x, \frac{t_{j+1}+t_j}{2} + \frac{h}{2}s) \varphi(\frac{t_{j+1}+t_j}{2} + \frac{h}{2}s) ds \quad (3)$$

```

$$
\begin{matrix}
\%перший рядок \\
a_{11} & a_{12} & a_{13} \\
\%другий рядок \\
a_{21} & a_{22} & a_{23} \\
\%третій рядок \\
a_{31} & a_{32} & a_{33}
\end{matrix}
$$

```

$$\begin{array}{ccc} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{array}$$
$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$
$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Оточення *vmatrix*:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

Оточення *Vmatrix*:

$$\left\| \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \right\|$$

Ряд точок у матрицях можна видрукувати командою `\hdotsfor`. Обов'язковий аргумент команди задає число стовпців, зайнятих точками, а необов'язковий, що стоїть першим – відстань між точками у відносних одиницях до стандартного. Наприклад:

```


$$K_n^a(\omega) =$$


$$\begin{vmatrix} a_{11} & 0 & \hdotsfor{2} & 0 \\ \omega_1 & a_{22} & 0 & \dots & 0 \\ \hdotsfor{2}[2]{5} \\ \omega_{n1} & \omega_{n2} & \dots & \omega_{n,n-1} & a_{nn} \end{vmatrix}$$


```

Результат компіляції:

$$K_n^a(\omega) = \begin{vmatrix} a_{11} & 0 & \dots & 0 \\ \omega_1 & a_{22} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \omega_{n1} & \omega_{n2} & \dots & \omega_{n,n-1} & a_{nn} \end{vmatrix}$$

Для вставки матриці в текст документу можна скористатися оточенням *smallmatrix* (розміщується в межах текстової формули). Наприклад (матриця вставляється між фігурними дужками, створених командами `\left(` та `\right)`):

Приклад ілюструє можливість вставки матриці $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right)$ в текстовий рядок.

Результат компіляції:

Приклад ілюструє можливість вставки матриці $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ в текстовий рядок.

Система умов з дужками

Для набору систем умов в пакеті *amsmath* передбачено оточення *cases*. Наприклад:

```


$$f(x) =$$


$$\begin{cases} \sin(\omega t + \varphi) & \text{при } x < 0 \\ \cos(\omega t + \varphi) & \text{при } x > 0 \end{cases}$$


```

Результат компіляції:

$$f(x) = \begin{cases} A \sin(\omega t + \varphi) & \text{при } x < 0, \\ 1 & \text{для } x = 0, \\ A \cos(\omega t + \varphi) & \text{при } x > 0 \end{cases}$$

Кратні інтеграли

Команди `\iint`, `\iiint` дозволяють створювати кратні інтегралів, а команда `\idotsint` виводу знаки інтегралу із багатокрапкою між ними: команди визначені в пакеті *amsmath*. Наприклад:

$$\iint\limits_S f(x,y) \, dx dy$$

Результат компіляції:

$$\iint_S f(x,y) \, dx dy$$

$$\iiint\limits_V f(x,y,z) \, dx dy dz$$

Результат компіляції:

$$\iiint_V f(x,y,z) \, dx dy dz$$

$$\idotsint\limits_V f(x,y,z) \, dx dy dz$$

Результат компіляції:

$$\int \cdots \int_V f(x, y, z) dx dy dz$$

Багаторядкові індекси

Команда `\substack` з пакету `amsmath` дозволяє створювати багаторядкові індекси для символів із межами (див. додаток №5). Вони розбиваються на рядки командою `\\`. Наприклад:

```


$$\sum_{\substack{0 \leq i < N \\ 0 \leq j < N \\ 0 \leq k < N}} S_0(i, j, k)$$


```

Результат компіляції:

$$\sum_{\substack{0 \leq i < N \\ 0 \leq j < N \\ 0 \leq k < N}} S_0(i, j, k)$$

Індекси по кутах символів

Команда `\sideset` (пакету `amsmath`) дозволяє виводити індекси по кутах символів із межами (див. додаток №5). Наприклад:

```


$$\prod_{i=1}^2 \prod_{k=1}^j D_k$$


```

Результат компіляції:

$$\prod_{i=1}^2 \prod_{k=1}^j D_k$$

Стрілки з індексами

Команди `\xrightarrow` та `\xrightarrow` з пакету `amsmath` дозволяють створювати стрілки з індексами. Необов'язковий параметр задає індекси під стрілкою, а обов'язковий – над нею. Наприклад:

```


$$\Omega \oplus \Psi \xrightarrow[\alpha]{\omega} H$$


```

Результат компіляції:

$$-\infty \xleftarrow[k]{x=1} \Omega \oplus \Psi \xrightarrow{\Delta_k \alpha(\omega)} H$$

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скомпілюйте їх та перегляньте результат.
2. Відповідно до індивідуального завдання наберіть виключні формули із автоматичною нумерацією.
3. Використовуючи оточення `eqnarray` самостійно створіть на окремій сторінці документу багаторядкову формулу.
4. На новій сторінці введіть фрагмент тексту (власний), де використайте генерацію посилань на раніше створені формули та сторінки, на які вони попадають.

Завдання для індивідуального виконання

№ варіанту	Завдання	
1	2	
1.	$\lim_{n \rightarrow \infty} n(\sqrt{n^2 + 2} - \sqrt{n^2 - 2})$ $\lim_{x \rightarrow 0} \frac{e^{\sin 2x} - e^{\sin 3x}}{\operatorname{tg} 2x}$ $\begin{cases} x = e^t \sin t, \\ y = e^t \cos t \end{cases}$ $\vec{a} = (x - 2y)\vec{i} + (yz + x)\vec{j} + z^2\vec{k}.$	$\int \frac{3x^3 + 11x^2 + 16x + 10}{(x+1)^2(x^2 + x + 2)} dx$ $\int_a^b f(x) dx = F(b) - F(a) = F(x) \Big _a^b$ $\sum_{n=1}^m \frac{(-1)^{n-1} (x-1)^n}{2^{n^2}}$
2.	$\lim_{n \rightarrow \infty} n(\sqrt{n(n-1)} - \sqrt{n^2 + 2})$ $\lim_{x \rightarrow 0} \frac{2^x + 2^{-x} - 2}{1 - \cos 3x}$ $\begin{cases} x = \sqrt{t-2}, \\ y = \frac{(t+1)\sqrt{t-2}}{\sqrt{t-2}} \end{cases}$ $\vec{a} = y\vec{i} + y^2\vec{z}\vec{j} - 2x\vec{k},$	$\int \frac{3x^3 - 2x^2 + x + 1}{(x-1)^2(x^2 + x + 1)} dx$ $\int_a^b f(x) dx = \int_a^b f(\varphi(t))\varphi'(t) dt$ $\sum_{n=1}^{\infty} \left(1 - \frac{1}{n}\right)^{n^2} (x-1)^n$

1	2
3.	$\lim_{n \rightarrow \infty} (n - \sqrt[3]{n^3 - 4}) \cdot n\sqrt{n}$ $\lim_{x \rightarrow -1} \frac{x^2 - 1}{\sin 2\pi x}$ $\begin{cases} x = \sin t + \cos t, \\ y = \cos 2t \end{cases}$ $\vec{a} = (y - x)\vec{i} + 2x\vec{j} + 2z^2\vec{k},$ <p>Створіть матрицю розміром 4 x 5 (з квадратними дужками)</p>
4.	$\lim_{n \rightarrow \infty} (\sqrt{(n^2 + 2)(n^2 - 3)} - \sqrt{n^4 - 8})$ $\lim_{x \rightarrow 0} \frac{\sqrt{1 + \sin 2x} - \sqrt{1 + \operatorname{tg} 2x}}{x^3}$ $\begin{cases} x = \sin^2 t, \\ y = \operatorname{ctg}^2 t \end{cases}$ $\vec{a} = (z - yx)\vec{i} + (2y^2 + 1)\vec{j} + (x - z^2)\vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з круглими дужками)</p>
5.	$\lim_{n \rightarrow \infty} \frac{\sqrt{n^5 - 6} - n\sqrt{n(n^2 + 4)}}{\sqrt{n}}$ $\lim_{x \rightarrow 0} \frac{1 - \sqrt{1 - x \sin 5x}}{e^{2x^2} - 1}$ $\begin{cases} x = \sin t, \\ y = \cos^3(t/2) \end{cases}$ $\vec{a} = (y + xz)\vec{i} + 2x\vec{j} + (z^2 - xy)\vec{k},$ <p>Створіть матрицю розміром 4 x 5 (з круглими дужками)</p>
6.	$\lim_{n \rightarrow \infty} (\sqrt{n(n+3)} - \sqrt{n^2 - 3n + 2})$ $\lim_{x \rightarrow -3} \frac{1 + \sqrt[3]{2+x}}{\sin \pi x}$ $\begin{cases} x = \sqrt{t+1}, \\ y = \sqrt[3]{t/(t+1)} \end{cases}$ $\vec{a} = xy\vec{i} + yz\vec{j} + xz\vec{k},$ <p>Створіть матрицю розміром 3 x 5 (з круглими дужками)</p>

1	2
7.	$\lim_{n \rightarrow \infty} (\sqrt{(n+4)(n-1)} - \sqrt{(n+7)(n-4)})$ $\lim_{x \rightarrow 0} \frac{1 - \sqrt{\cos 2x}}{\sin^2 3x}$ $\begin{cases} x = \ln t, \\ y = \operatorname{arctg} t \end{cases}$ $\vec{a} = (x^2 - 2y)\vec{i} + (yz - x)\vec{j} + xyz\vec{k},$ <p>Створіть матрицю розміром 1 x 3 (з квадратними дужками)</p>
8.	$\lim_{n \rightarrow \infty} n(\sqrt{n(n^4 - 4)} - \sqrt{n^5 + 1})$ $\lim_{x \rightarrow 0} \frac{4^{x+1} - 4}{\ln(1 - x\sqrt{1 + xe^x})}$ $\begin{cases} x = \sqrt[3]{t-1}, \\ y = t/\sqrt{t-1} \end{cases}$ $\vec{a} = (y - \sin z)\vec{i} + (2x - y^2)\vec{j} + (x - 2z)\vec{k},$ <p>Створіть матрицю розміром 3 x 4 (з квадратними дужками)</p>
9.	$\lim_{n \rightarrow \infty} n(\sqrt[3]{2 + 27n^3} - 3n)$ $\lim_{x \rightarrow 2} \frac{2 - x}{\lg(x-1)}$ $\begin{cases} x = \arcsin t, \\ y = \sqrt{1-t^2} \end{cases}$ $\vec{a} = (y - z)\vec{i} + xyz\vec{j} + (x - y)\vec{k},$ <p>Створіть матрицю розміром 3 x 5 (з квадратними дужками)</p>
10.	$\lim_{n \rightarrow \infty} (\sqrt[3]{(n+1)^3} - \sqrt[4]{(n-1)^3})$ $\lim_{x \rightarrow -2} \frac{x^3 + 8}{\sin \pi(x+2)}$ $\begin{cases} x = \cos 3t, \\ y = \ln \sin 3t \end{cases}$ $\vec{a} = (2zx - 3z)\vec{i} + e^y(x+z)\vec{j} + (x^2 - x)\vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з круглими дужками)</p>

1	2
11.	$\lim_{n \rightarrow \infty} \sqrt{n}(\sqrt{n+3} - \sqrt{n-2})$ $\lim_{x \rightarrow 0} \frac{1 - \sin x^2 - \cos 3x}{\sin^2 x}$ $\begin{cases} x = \operatorname{ch}^2 t, \\ y = 1/\operatorname{sh}^2 t \end{cases}$ $\vec{a} = e^{xy} \vec{i} + (x^2 - z) \vec{j} + (z^2 + y) \vec{k},$ <p>Створіть матрицю розміром 3 x 4 (з круглими дужками)</p>
12.	$\lim_{n \rightarrow \infty} \frac{\sqrt{n(n^3 + 3)} - \sqrt{(n^4 + 1)(n^2 - 2)}}{2n}$ $\lim_{x \rightarrow 2} \frac{\operatorname{tg} x - \operatorname{tg} 2}{\ln x - \ln 2}$ $\begin{cases} x = 1/t^2, \\ y = 1/(4t^2 + 1) \end{cases}$ $\vec{a} = (\sqrt[3]{3x + 3y}) \vec{i} + (y^2 - 2z) \vec{j} + (x + y + \sqrt[3]{2z}) \vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з круглими дужками)</p>
13.	$\lim_{n \rightarrow \infty} \sqrt{n+1}(\sqrt{n+5} - \sqrt{n+3})$ $\lim_{x \rightarrow 0} \frac{\sin 3x - 3 \sin x}{x \ln \cos 2x}$ $\begin{cases} x = \sqrt{t-2}, \\ y = \ln(t-1) \end{cases}$ $\vec{a} = \sqrt{x} \vec{i} - (\sqrt[3]{xy} + z) \vec{j} + (\sqrt[3]{2xz} + 2y) \vec{k},$ <p>Створіть матрицю розміром 3 x 3 (з квадратними дужками)</p>
14.	$\lim_{n \rightarrow \infty} (\sqrt{9n^2 + 2n + 1} - 3n)$ $\lim_{x \rightarrow 0} \frac{\operatorname{tg}^2 x - \cos x + 1}{\sin^2 3x}$ $\begin{cases} x = \sqrt[3]{t+1}, \\ y = \ln t \end{cases}$ $\vec{a} = \left(y - \frac{x^2}{z^2 + 1}\right) \vec{i} - \left(x + \frac{y^2}{z^2 + 1}\right) \vec{j} + \frac{z^2}{x^2 + y^2 + 1} \vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з квадратними дужками)</p>

1	2
15.	$\lim_{n \rightarrow \infty} \sqrt{n^3 + 7}(\sqrt{n^3 + 3} - \sqrt{n^3 - 2})$ $\lim_{x \rightarrow \pi/2} \frac{\ln \sin x}{\sin(x - \pi/4) - \cos(x - \pi/4)}$ $\begin{cases} x = \sqrt{t-2}, \\ y = (t+1)/\sqrt{t-2} \end{cases}$ $\vec{a} = \sqrt{x^2 + 1} \vec{i} + (x + 2z) \vec{j} + (2x + y + z) \vec{k},$ <p>Створіть матрицю розміром 5 x 3 (з квадратними дужками)</p>
16.	$\lim_{n \rightarrow \infty} (\sqrt{(n^2 + 2)(n^2 + 3)} - \sqrt{(n^2 - 2)(n^2 - 3)})$ $\lim_{x \rightarrow \pi/4} \frac{\cos^2 2x}{1 - \sin^2 2x}$ $\begin{cases} x = \cos t, \\ y = \operatorname{cosec} t \end{cases}$ $\vec{a} = (\ln x + yz) \vec{i} - (3z + 1) \vec{j} + (\sqrt{x^2 z^2 + 1} + y) \vec{k},$ <p>Створіть матрицю розміром 4 x 5 (з круглими дужками)</p>
17.	$\lim_{n \rightarrow \infty} \frac{\sqrt{n(n^4 + 3)} - \sqrt{(n^3 + 1)(n^2 - 2)}}{\sqrt{n}}$ $\lim_{x \rightarrow 5} \frac{\operatorname{tg} \pi x}{1 - \lg 2x}$ $\begin{cases} x = \sqrt{t-1}, \\ y = \sqrt[3]{t} \end{cases}$ $\vec{a} = (\cos x - y) \vec{i} - (\sin yz + x) \vec{j} + (\sin xy + z) \vec{k},$ <p>Створіть матрицю розміром 4 x 4 (з прямими подвійними дужками)</p>
18.	$\lim_{n \rightarrow \infty} (\sqrt{n} \sqrt{n-2} - \sqrt{(n+1)(n^2 - 3n + 1)})$ $\lim_{x \rightarrow 1} \frac{3^x - 3}{\sin(1 - x^2)}$ $\begin{cases} x = t + \sin 3t, \\ y = 2 + 3 \cos 3t \end{cases}$ $\vec{a} = (z^2 - 2y) \vec{i} + (y^2 - z) \vec{j} - (2x^2 - y) \vec{k},$ <p>Створіть матрицю розміром 5 x 4 (з прямими дужками)</p>

1	2
19.	$\lim_{n \rightarrow \infty} \frac{\sqrt{(n^4 - 2)(n^2 + 1)} - \sqrt{n^6 - 3}}{n}$ $\lim_{x \rightarrow \frac{\pi}{6}} \frac{\cos(\pi - 3x)}{1 - 2 \sin x}$ $\begin{cases} x = t - \sin 2t, \\ y = 3 - 2 \cos 2t \end{cases}$ $\vec{a} = (xz + y)\vec{i} - (yz + x)\vec{j} + (xy + 2z)\vec{k},$ <p>Створіть матрицю розміром 3 x 5 (з квадратними дужками)</p>
20.	$\lim_{n \rightarrow \infty} \sqrt[3]{n}(\sqrt{n^2 + 4} - \sqrt[3]{n(n-2)})$ $\lim_{x \rightarrow 0} \frac{\sqrt{x+4} - 2}{\sin 2x}$ $\begin{cases} x = \sin^2 t, \\ y = \operatorname{ctg}^2 t \end{cases}$ $\vec{a} = \frac{x^2 + y^2}{2} \vec{i} + (xy + z)\vec{j} - (y + xz)\vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з круглими дужками)</p>
21.	$\lim_{n \rightarrow \infty} (\sqrt[3]{(n^2 + 3)(n^2 - 1)} - n)$ $\lim_{x \rightarrow -1} \frac{2^{x+1} - 1}{\ln(2+x)}$ $\begin{cases} x = e^t \sin t, \\ y = e^t \cos t \end{cases}$ $\vec{a} = \left(\frac{2xz}{y} - z \right) \vec{i} - \frac{x+z}{y^2+1} \vec{j} + \left(\frac{x^2}{y} + x \right) \vec{k},$ <p>Створіть матрицю розміром 4 x 4 (з прямими подвійними дужками)</p>
22.	$\lim_{n \rightarrow \infty} n(\sqrt[4]{n^4 + 2} - \sqrt[4]{n^4 - 1})$ $\lim_{x \rightarrow \frac{\pi}{4}} \frac{2 \sin^2 x - \sin 2x}{\operatorname{Intg} x}$ $\begin{cases} x = \operatorname{ctg} t, \\ y = \frac{1}{\sin t} \end{cases}$ $\vec{a} = e^{yz} \vec{i} + (e^x z^2 - 2z)\vec{j} + (2yze^x + x)\vec{k},$ <p>Створіть матрицю розміром 4 x 3 (з прямими дужками)</p>

1	2
23.	$\lim_{n \rightarrow \infty} n^2(\sqrt[3]{n^3 + 3} - \sqrt[3]{n^3 - 1})$ $\lim_{x \rightarrow 2} \frac{\sin \pi x}{\ln(3x - 5)}$ $\begin{cases} x = \sqrt[3]{t}, \\ y = \sqrt[3]{1-t} \end{cases}$ $\vec{a} = (y^2 z + 2y)\vec{i} + (2xyz + z^2)\vec{j} + (x^2 y + z)\vec{k},$ <p>Створіть матрицю розміром 2 x 3 (з круглими дужками)</p>
24.	$\lim_{n \rightarrow \infty} (n + \sqrt[3]{3 - n^3})$ $\lim_{x \rightarrow \pi} \frac{\sin^2 x}{\ln(1 + \cos^2 \frac{x}{2})}$ $\begin{cases} x = \sqrt{t}, \\ y = \frac{1}{\sqrt{1-2t}} \end{cases}$ $\vec{a} = (x^2 z^2 - y)\vec{i} + (y^2 z^2 + x)\vec{j} + (x^2 y^2 + z)\vec{k},$ <p>Створіть матрицю розміром 4 x 2 (з квадратними дужками)</p>
25.	$\lim_{n \rightarrow \infty} (\sqrt{n^2 - 5n + 6} - n)$ $\lim_{x \rightarrow \frac{\pi}{4}} (\sin 2x)^{\cos^{-1}(3x - \frac{\pi}{4})}$ $\begin{cases} x = \sin t, \\ y = \cos^3(\frac{t}{2}) \end{cases}$ $\vec{a} = (x^2 - z)\vec{i} + (x + 5z)\vec{j} + (2y + \sin z)\vec{k},$ <p>Створіть матрицю розміром 2 x 4 (з круглими дужками)</p>
26.	$\lim_{n \rightarrow \infty} \left(\frac{7n^2 + 2n - 2}{7n^2 + n + 1} \right)^{-3n}$ $\lim_{x \rightarrow 1} (3 - 2x)^{\frac{\ln(3+2x)}{\ln(3-2x)}}$ $\begin{cases} x = e^t, \\ y = \arccos t \end{cases}$ $\vec{F} = (x^2 - y^2)\vec{i} + (x^2 + 2y^2)\vec{j},$ <p>Створіть матрицю розміром 4 x 5 (з квадратними дужками)</p>

Контрольні питання

1	2
27.	$\lim_{n \rightarrow \infty} \left(\frac{n^3 - 5}{n^3 + n} \right)^{1-n^2} \quad \int (2x^3 + 1) \operatorname{ch} 2x dx$ $\lim_{x \rightarrow \pi/2} \left(\operatorname{ctg} \frac{x}{2} \right)^{\operatorname{tg} x} \quad \int_{\pi/4}^{\pi/2} x^2 \cos 3x dx$ $\begin{cases} x = \sin 2t, \\ y = \ln \cos 2t \end{cases} \quad \sum_{n=1}^{\infty} \frac{5^n x^{4n} \cos(2x - \pi n)}{\sqrt{2n}}$ $\vec{F} = (2xy + y)\vec{i} + (x^2 + xy)\vec{j},$ <p>Створіть матрицю розміром 3 x 4 (з прямими дужками)</p>
28.	$\lim_{n \rightarrow \infty} \left(\frac{2n^2 + n + 2}{2n^2 + n - 1} \right)^{n-n^2} \quad \int \frac{\ln^2 3x}{x^2 \cdot \sqrt[3]{2x}} dx$ $\lim_{x \rightarrow 0} (1 - \sqrt{x})^{\operatorname{tg} x / \sin x} \quad \int_{\pi/2}^{\pi} x \sin^3 x dx$ $\begin{cases} x = \operatorname{sh} t, \\ y = \operatorname{ch}^{2/3} t \end{cases} \quad \sum_{n=1}^{\infty} \frac{4^n x^{2n} \cos(2x + \pi n)}{\sqrt[3]{n}}$ $\vec{F} = (x + y\sqrt{x^2 + y^2})\vec{i} + (y - x\sqrt{x^2 + y^2})\vec{j},$ <p>Створіть матрицю розміром 6 x 2 (з прямими подвійними дужками)</p>
29.	$\lim_{n \rightarrow \infty} \left(\frac{2n^2 - n + 3}{2n^2 + 2n + 1} \right)^{-n+1} \quad \int \frac{\arccos 2x}{\sqrt{1+2x}} dx$ $\lim_{x \rightarrow \pi} \left(\operatorname{tg} \frac{x}{4} \right)^{\sec(x - \frac{3\pi}{2})} \quad \int_{-2}^1 \frac{x dx}{x^3 + 1}$ $\begin{cases} x = t \sin t + \cos t, \\ y = \sin t - t \cos t \end{cases} \quad \sum_{n=1}^{\infty} \frac{9^n x^{2n} \sin(3x + \pi n)}{\sqrt{n}}$ $\vec{F} = \frac{y}{1+x^2+y^2} \vec{i} + \frac{x}{1-x^2-y^2} \vec{j},$ <p>Створіть матрицю розміром 4 x 5 (з круглими дужками)</p>
30.	$\lim_{n \rightarrow \infty} \left(\frac{3n^2 + 2n + 1}{3n^2 - n + 1} \right)^{-n^2} \quad \int (2x^3 + 1) \sin 3x dx$ $\lim_{x \rightarrow \pi/2} \left(\operatorname{ctg} \frac{x}{2} \right)^{\operatorname{tg} x} \quad \int_1^2 x^{5/2} \sqrt{2-x} dx$ $\begin{cases} x = \sin t + \cos t, \\ y = \cos 2t \end{cases} \quad \sum_{n=1}^{\infty} \frac{2^{4n} x^{2n} \sin(x + \pi n)}{n}$ $\vec{F} = (x^2 + y^2)\vec{i} + (x^2 - y^2)\vec{j},$ <p>Створіть матрицю розміром 7 x 2 (з квадратними дужками)</p>

1. Перерахуйте можливості оформлення математичних формул за допомогою оточення **equation**.
2. Як пронумерувати формули в ручному режимі?
3. Як задати автоматичну нумерацію формул в документі?
4. Як створити в тексті посилання на формулу та номер сторінки, де фона знаходиться?
5. Як створити систему рівнянь?
6. Перерахуйте основні типи вирівнювання систем рівнянь.
7. Як ввести довгу формулу та пронумерувати її?
8. Які додаткові можливості застосування індексів дає оточення **ams-math**?
9. Як правильно ввести стандартні математичні функції?
10. Як створити матрицю?
11. Які типи дужок можна використати при створенні матриці?
12. Як ввести кратні інтеграли?
13. Як створити систему умов з дужкою?
14. Які додаткові можливості створення стрілок дає оточення **ams-math**?

Лабораторна робота №7

Тема роботи: Робота з таблицями у видавничій системі LaTeX

Мета роботи: верстка документу з використанням простих та складних таблиць

Для виконання роботи необхідно знати:

- загальну структуру документу LaTeX;
- використання оточення **tabbing** для створення простих таблиць;
- створення таблиць у оточенні **tabular**;
- робота з таблицею як із «плаваючим об'єктом»;
- призначення та можливості оточення **array**.

Теоретичні відомості

В LaTeX для побудови таблиць може бути використано одне із наступних оточень:

- **tabbing**: створення таблиці в машинописному стилі;
- **tabular**: універсальний засіб створення таблиць будь-яких стилів;
- **table**: створення таблиці як «плаваючого» об'єкту;
- **array**: створення таблиць в математичному стилі, зокрема векторів та масивів.

Оточення **tabbing**

Оточення використовується для створення відносно простих таблиць, якщо ширина стовпців відома наперед (див. приклад нижче). Таблиця формується в ручному режимі: розбиття на стовпчики задається за допомогою керуючого символу **\=** (задання позицій табуляції) у першому рядку таблиці. Далі порядково формуються комірки таблиці за допомогою команди **\>**. Перехід на наступний рядок задається командою ****. Вона може мати необов'язковий параметр для задання додаткової відстані між рядками таблиці. Приклад **\\[10pt]** задає додаткову відстань між рядками в 10pt.

Оточення формує наповнення комірок тільки в один рядок форматovanого тексту і не дозволяє використовувати форматовані абзаци із кількох рядків та перенесенням слів. Текст, що знаходиться між двома

командами **\>** чи **\=** утворює групу, тому при його форматуванні в межах комірки немає потреби використовувати фігурні дужки.

Для створення шаблону (розбиття на комірки) таблиці у першому рядку вводиться будь-який текст необхідної ширини, а сам рядок не відображається за допомогою команди **\kill**. Наприклад:

```
\begin{tabbing}
% створення шаблону таблиці
AAA \= AAAAAAAAAAAAAAAAAA
      \= AAAAAAAAAAAAAAAAAA\= \kill
% формування наповнення таблиці
\bf № \> \bf Автор \> \bf Кількість
примірників\\[10pt]
1 \> Львовский С.М. \> 2\\
2 \> Плис А. И. \> 3\\
3 \> Кнут Д. \> 12\\[10pt]
  \> \bf Всього \> \it 17\\
\end{tabbing}
```

Результат компіляції:

№	Автор	Кількість примірників
1	Львовский С.М.	2
2	Плис А. И.	3
3	Кнут Д.	12
Всього		17

Основним недоліком оточення є те, що при некоректному розбитті на стовпці можливе наповнення тексту сусідніх комірок один на одного (текст не поміщається в межах заданої ширини комірок). Наприклад (для першого автора кількість примірників «наповзає» на прізвище):

```
\begin{tabbing}
AAA \= AAAAAAA \= AAAAAAA\= \kill
\bf № \> \bf Автор \> \bf Кількість
примірників\\[10pt]
1 \> Львовский С.М. \> 2\\
2 \> Плис А. И. \> 3\\
3 \> Кнут Д. \> 12\\[10pt]
  \> \bf Всього \> \it 17\\
\end{tabbing}
```

Результат компіляції:

№	Автор	Кількість примірників
1	Львовский С.М.	
2	Пліс А. И.	3
3	Кнут Д.	12
Всього		17

Суттєвим обмеженням є те, що для створення складнішої структури таблиці недопустимим є вкладення одного оточення **tabbing** у інше.

Оточення **tabular**

Оточення є більш гнучким та універсальним засобом створення таблиць. В ньому ширина колонок вибирається автоматично за максимальної шириною їх вмісту. Основними елементами оточення є:

преамбула таблиці – задається як обов'язків параметр та визначає загальне оформлення. В найпростішому випадку це набір літер (по одній на колонку), що описують структуру таблиці:

Літера	Особливості форматування тексту
l	Вирівнювання в колонці по лівому краю.
r	Вирівнювання в колонці по правому краю.
c	Вирівнювання в колонці по центру.
 	Використовується для розділення стовпців таблиці вертикальними лініями.
@{символ}	Вставка між комірками будь-якого символу (символ), заданого як обов'язковий параметр.
p{N}	Текст у комірці буде розміщено у кілька рядків, якщо він має більшу ніж комірка ширину. N задає ширину стовпця з використанням будь-яких допустимих одиниць довжини.

Розміщенням таблиці відносно тексту (якщо вона знаходиться в межах тексту) можна управляти за допомогою наступних необов'язкових параметрів у преамбулі таблиці:

Літера	Особливості вирівнювання таблиці в тексті
t	Вирівнювання по верхньому рядку – верхній рядок таблиці буде розміщено на одному рівні із текстовим рядком.
b	Вирівнювання по нижньому рядку – нижній рядок таблиці буде розміщено на одному рівні із текстовим рядком.
c	Вирівнювання по центру - середина таблиці буде розміщена на одному рівні із текстовим рядком.

Ширина комірок визначається автоматично за заданим форматом. Висота комірок визначається найбільшою висотою введенного тексту у рядку. Приклад формування преамбули таблиці:

```
\begin{tabular}[c]{|c|c|p{300pt}|}
```

Тут необов'язковий параметр **[c]** задає розміщення таблиці у тексті документу. Преамбула формує таблицю, що складається із трьох стовпців: текст в перших двох комірках буде вирівняно по центру, а для третьої комірки явним чином задано ширину **300pt**. Таблиця зліва та справа буде обведена подвійною лінією, а комірки будуть розділені одинарними лініями.

Власне таблиця створюється за допомогою управляючих символів:

& - формування рядків таблиці (поділ на комірки);

**** - формування стовпців таблиці. Команда може мати необов'язковий параметр, що задає додатковий зсув по вертикалі. Наприклад: **\\[10pt]** задає додатковий зсув між рядками таблиці величиною **10pt**.

Для форматування таблиці додатково можна використати наступні команди:

\hline – горизонтальна розділююча лінія на всю ширину таблиці;

\cline – горизонтальна лінія тільки в межах заданих номерів комірок. Наприклад **\cline{2-3}** проводить лінію тільки для другої та третьої комірки;

\multicolumn – об'єднання кількох суміжних комірок рядка. Команда має три обов'язкових параметри, що задаються у окремих фігурних дужках:

```
\multicolumn{число}{преамбула}{текст}
```

число - кількість комірок, що об'єднуються;

преамбула - преамбула комірки – аналогічна до преамбули таблиці;

текст - текст, що вводиться в даній комірці.

Для об'єднання комірок по вертикалі необхідно підключити в преамбулі документу пакет **multirow**. У пакеті визначено команду об'єднання комірок:

\multirow – об'єднання кількох суміжних комірок стовпчика (по вертикалі):

\multirow{число}{ширина}{текст}

число - кількість комірок, що об'єднуються;

ширина - ширина об'єднаної комірки. Якщо явно не задана – обчислюється автоматично;

текст - текст, що виводиться в даній комірці.

Приклад використання команд об'єднання комірок для формування складної структури таблиці:

```
\begin{tabular}[c]{|c|c|c|p{100pt}|}
\hline
% об'єднання комірок для формування заголовку
% таблиці
\multicolumn{4}{|c|}{Заголовок } \\[2pt]
\hline
1&2 &\multicolumn{2}{|c|}{Підзаголовок } \[
\hline
\multirow{2}{40pt}{текст1} & &
& \multirow{4}{40pt}{текст2} \[
\cline{2-3}
& \multirow{3}{50pt}{текст3} & & \[
\cline{1-1} \cline{3-3}
\multirow{2}{40pt}{текст4}& & & \[
\cline{3-3}
& & & \[
\hline
\end{tabular}
```

Результат компіляції:

Заголовок			
1	2	Підзаголовок	
текст1	текст3		текст2
текст4			

Щоби уникнути напівзаповнених сторінок таблиця в тексті повинна «плавати» - міняти своє положення в межах сторінки, або переноситися на наступну. Для роботи із таблицею як із плаваючим об'єктом, формування заголовку та її автоматичної нумерації служить оточення **table**. Для створення таблиці у ньому необхідно додатково використати одне із розглянутих вище оточень (**tabbing** або **tabular**) за наступним синтаксисом:

```
\begin{table}[розміщення]
\begin{tabular}...
...
\end{tabular}...
\caption{підпис до таблиці}
\label{мітка таблиці}
```

Необов'язковий параметр **[розміщення]** визначає бажане розміщення таблиці на сторінці:

h - в місці знаходження оточення **table** в тексті;

t - вверху сторінки;

b - внизу сторінки;

p - на окремій сторінці, яка призначена для «плаваючих» об'єктів.

Одночасно можна використовувати кілька параметрів. Для стандартного класу документів **article** за замовчуванням приймається значення параметрів **tblp** – спочатку LaTeX пробує розмістити таблицю вгорі сторінки, якщо це не можливо – то внизу, а якщо і це неможливо – таблиця переноситься на окрему сторінку для «плаваючих» об'єктів.

Пакет **babel** із опцією **ukrainian** визначає ключове слово для формування заголовка таблиці Табл. Його можна переозначити на початку документу за допомогою команди (див. Додаток №4):

```
\renewcommand {\tablename}{Таблиця}
```

Команда задає для таблиці новий заголовок **Таблиця**.

Додатково управляти стилем заголовка таблиці, якщо в преамбулі документу підключити пакет **caption**, можна за допомогою команди:

```
\captionsetup[table]{параметри}
```

параметри через кому задають:

font=шрифт – шрифт, яким відображається заголовок таблиці. Наприклад **font=sf**;

labelsep=символ – тип розділового знаку між ключовим словом та заголовком таблиці.

Таблиця №15. Типи розділових знаків, що можуть використовуватися при формуванні заголовку

Символ	Розділовий знак
<code>none</code>	Розділовий знак відсутній
<code>colon</code>	Двокрапка
<code>period</code>	Крапка
<code>space</code>	Пробіл
<code>quad</code>	Пробіл величиною в чотири символи
<code>newline</code>	Ключове слово та заголовок знаходяться в різних рядках

Положення заголовку та номери таблиці (вгорі чи внизу таблиці) визначається положенням команди `\caption` відносно оточення `tabular` (до чи після оточення).

Приклад створення таблиці із використанням основних управляючих конструкцій, описаних вище:

```
\documentclass[12pt,a4paper]{article}
\usepackage{cp1251}{inputenc}
\usepackage[ukrainian]{babel}
\usepackage{caption}
\captionsetup[table]{font=sf, labelsep=newline}
\begin{document}
\renewcommand{\tablename}{Таблиця}
```

Документ ілюструє основні прийоми створення таблиці засобами видавничої системи.

Використано основні прийоми форматування наповнення таблиці, її розміщення, проведення ліній та формування заголовку таблиці

```
\begin{table}[t]
\begin{tabular}[c]{|c||c||c||p{300pt}|}
\hline \hline
\multicolumn{3}{|c|}{\bf{Обов'язкові
параметри оточення tabular}} } \\\[10pt]
\hline
\hline
```

```
\& \it{Параметр} & \it{Призначення} \\\[10pt]
\hline
1 & l & Введені в комірку дані вирівнюються в
колонці по лівому краю \\\[10pt]
\hline
2 & r & Вирівнювання в колонці по правому краю \\\[10pt]
\hline
3 & c & Вирівнювання в колонці по центру \\\[10pt]
\hline
4 & | & Символ використовується для розділення
стовпців таблиці вертикальними лініями. \\\[10pt]
\cline{2-3}
& || & При необхідності проведення кілька
вертикальних ліній необхідно поставити відповідну
кількість символів. У даному випадку буде проведено
дві вертикальні лінії \\\[10pt]
\hline \hline
\end{tabular}
\caption{Таблиця основних обов'язкових параметрів}
\label{tbl1}
\end{table}
\end{document}
```

Результат компіляції:

Обов'язкові параметри оточення tabular		
№	Параметр	Призначення
1	l	Введені в комірку дані вирівнюються в колонці по лівому краю
2	r	Вирівнювання в колонці по правому краю
3	c	Вирівнювання в колонці по центру
4		Символ використовується для розділення стовпців таблиці вертикальними лініями.
		При необхідності проведення кілька вертикальних ліній необхідно поставити відповідну кількість символів. У даному випадку буде проведено дві вертикальні лінії

Таблиця 1
Таблиця основних обов'язкових параметрів

Документ ілюструє основні прийоми створення таблиці засобами видавничої системи.

Використано основні прийоми форматування наповнення таблиці, її розміщення, проведення ліній та формування заголовку таблиці

Оточення `array`

Оточення за функціональними можливостями фактично повністю повторює оточення `tabular`, але використовується для роботи в математичному режимі: для створення векторів, матриць, системи рівнянь чи розбиття формул на декілька рядків:

```
\begin{array}{преамбула}
...
\end{array}
```

преамбула описує структуру таблиці аналогічно до оточення `tabular`.

В доповнення до стандартних типів стовпців, що використовується в попередньому оточенні, можна використовувати два нові:

m{N} – ширина комірки (аналогічно до **p{N}**) із одночасним вертикальним вирівнюванням вмісту по центру;

b{N} – ширина комірки із одночасним вирівнюванням вмісту по нижній базовій лінії останнього рядка.

Приклади використання оточення:

створення матриці:

```
$$
\begin{array}{cccc}
a_{11} & a_{12} & \dots & a_{1n} \\
a_{21} & a_{22} & \dots & a_{23} \\
\dots & \dots & \dots & \dots \\
a_{n1} & a_{n2} & \dots & a_{nn}
\end{array}
$$
```

Результат компіляції:

$$\begin{array}{cccc} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{23} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{array}$$

визначник матриці

```
$$
\left| \begin{array}{cc}
a_{11} & a_{12} \\
a_{21} & a_{22}
\end{array} \right|

```

```
\right|
$$
```

Результат компіляції:

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$$

система рівнянь:

```
$$
\left\{ \begin{array}{l}
a_1 x + b_1 y + c_1 z = k_1 \\
a_2 x + b_2 y + c_2 z = k_2 \\
a_3 x + b_3 y + c_3 z = k_3
\end{array} \right.
\right.

```

Зверніть увагу: щоби права фігурна дужка не відображалася після команди `\right` стоїть крапка.

Результат компіляції:

$$\begin{cases} a_1 x + b_1 y + c_1 z = k_1 \\ a_2 x + b_2 y + c_2 z = k_2 \\ a_3 x + b_3 y + c_3 z = k_3 \end{cases}$$

система умов:

```
$$
y(x) = \left\{ \begin{array}{l}
-1 \text{ при } x < 0 \\
0 \text{ при } x = 0 \\
1 \text{ при } x > 0
\end{array} \right.

```

Результат компіляції:

$$y(x) = \begin{cases} -1 & \text{при } x < 0 \\ 0 & \text{при } x = 0 \\ 1 & \text{при } x > 0 \end{cases}$$

багаторядкова формула (`\qquad` задає додатковий зсув по горизонталі для другого фрагменту формули):

```


$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$


```

Результат компіляції:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$

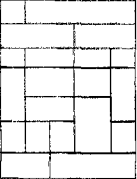
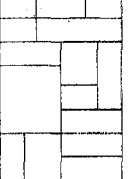

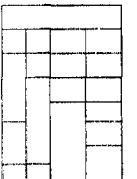
Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте їх та перегляньте результат.
2. Відповідно до індивідуального завдання створіть таблицю та заповніть її текстом на власний розсуд.
3. Використовуючи оточення `array` введіть власну багаторядкову формулу та систему рівнянь.

Завдання для індивідуального виконання

Варіант	Схема таблиці	Варіант	Схема таблиці
1	2	3	4
1.		11.	
2.		12.	

1	2	3	4
3.		13.	
4.		14.	
5.		15.	
6.		16.	
7.		17.	
8.		18.	

1	2	3	4
9.		19.	
10.		20.	

Контрольні питання

1. Які оточення можуть використовуватися в **LaTeX** для створення таблиць?
2. Як створити таблицю за допомогою оточення **tabbing**?
3. Як створити таблицю за допомогою оточення **tabular**?
4. Які обов'язкові параметри можна задати в преамбулі таблиці?
5. Яке призначення необов'язкових параметрів оточення **tabular**?
6. Які управляючі символи використовуються для формування наповнення таблиці?
7. Як створити таблицю як «плаваючий об'єкт»?
8. Як задати бажане розміщення таблиці на сторінці?
9. Як поміняти стиль заголовка до таблиці?
10. Як об'єднати суміжні рядки таблиці?
11. Як об'єднати суміжні стовпці таблиці?
12. Як провести розділові
13. лінії в таблиці?
14. Як переозначити ключове слово заголовка до таблиці?
15. Яке призначення оточення **array**?
16. Як за допомогою оточення **array** створити багаторядкову формулу?

Лабораторна робота №8

Тема роботи: Створення нових команд у видавничій системі **LaTeX**

Мета роботи: робота із макросами та створення нових оточень у видавничій системі **LaTeX**. Команди з аргументами

Для виконання роботи необхідно знати:

- порядок створення макросів;
- порядок створення нових оточень;
- призначення лічильників та їх використання;
- можливості та використання оточення типу «теорема».

Теоретичні відомості

Для спрощення набору складного тексту, особливо якщо він містить фрагменти, що повторюються, використовуються нові команди - макровизначення (макроси).

Створення нових макросів без аргументів

Для створення макросів використовується команда `\newcommand` і в найпростішому випадку вона має два обов'язкові аргументи:

`\newcommand{ \назва } { визначення }`

тут:

назва – ім'я макровизначення. Для імені використовується або послідовність англійських букв, або одна не буква. Ім'я не може повторювати вже існуючі в **LaTeX** команди;

визначення – фрагмент коду, що буде використовуватися при застосуванні макросу.

Макрос може бути визначений як в прсамбулі документу, так і в будь-якому місці документу. Якщо він визначений в межах групи чи оточення, то діє локально тільки в межах даної групи чи оточення.

Приклад визначення макросу із іменем **Tex**:

`\newcommand{ \Tex } {Приклад створення нової команди .}`

Визначення макросу може містити як звичайний текст, так і будь-який набір допустимих в **LaTeX команд.**

```
{\bf{Наприклад:}}

$$f(x)=ax^2+bx+c$$

}
```

Всюди в документі, де буде зустрічатись команда звернення до макросу `\TeX` буде виводитись фрагмент документу:

Приклад створення нової команди.

Визначення макросу може містити як звичайний текст, так і будь-який набір допустимих в \TeX команд.

Наприклад:

$$f(x) = ax^2 + bx + c$$

Створення нових макросів із аргументами

При необхідності передати макросу ряд аргументів (текст, що буде виводитись в заданому місці) використовується команда:

```
\newcommand{\назва}[число]{визначення}
```

тут:

число – кількість аргументів макросу. Максимально може бути передано до 9 аргументів. Місце вставки аргументів визначається мітками (**#1, #1, #1...**). Приклад створення макросу із двома аргументами:

```
\newcommand{\Zadacha}[2]{
{\bf Розділ #1}

{\it Задача #2}
}
```

При виклику макросу `\Zadacha{#1}{#2}` отримаємо:

Розділ №1
Задача №2

Цей-же макрос `\Zadacha{Системи рівнянь}{підвищеної складності}` виведе наступний текст:

Розділ Системи рівнянь
Задача підвищеної складності

При бажанні чи необхідності замінити вже існуючі команди \LaTeX на нові необхідно скористатися командою `\renewcommand`, синтаксис якої є аналогічний `\newcommand`. Команда `\providecommand` також дозволяє визначити нову команду, але при цьому якщо команди з таким іменем існує, то вона залишиться незмінною.

Створення нових оточень

Нове оточення створюється за допомогою команди:

```
\newenvironment{назва}[число]{перед}{після}
```

тут:

назва – ім'я оточення, яке формується аналогічно до формування імені нових команд;

число – кількість аргументів макросу. Максимально може бути передано до 9 аргументів. Може бути і відсутнім;

перед – аргумент, що обробляється до обробки тексту в середині оточення;

після – аргументу, що обробляється після обробки тексту в середині оточення.

Оточення без аргументів

Ряд аргументів у оточенні може бути і відсутнім. Наприклад (задається тільки назва оточення та аргумент, що обробляється до тексту оточення):

```
\newenvironment{Form}{\tt \Large}
```

При використанні наступного оточення в документі

```
\begin{Form}
```

Даний текст відформатовано за допомогою створеного нового оточення *Form*

```
\end{Form}
```

отримаємо текст великого розміру, надрукований шрифтом друкарської машинки:

Даний текст відформатовано за допомогою створеного нового оточення *Form*

Оточення із аргументами

Створення оточення із двома аргументами:

```
\newenvironment{Theor}[2]{\tt Теорема #1.#2.}
\large{}
```

При його використанні в документі:

```
\begin{Theor}[2]{5}
```

Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2+b^2=c^2$$

```
\end{Theor}
```

отримаємо:

Теорема 2.5. Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Вже існуючі оточення можуть бути переозначені за допомогою команди `\renewenvironment`. Синтаксис команди аналогічний `\newenvironment`.

Лічильники

Лічильники використовують для організації автоматичної нумерації об'єктів (створених за допомогою нових команд чи нових оточень), як це зроблено в **LaTeX** для нумерації розділів, підрозділів і т.д. Лічильник – змінна, яка може приймати цілі значення (як додатні, так і від'ємні). Лічильнику можна присвоїти будь-яке ціле значення та організовувати з їх допомогою автоматичну генерацію посилань.

Лічильник створюється за допомогою команди:

```
\newcounter{ім'я лічильника}
```

ім'я лічильника - формується аналогічно до формування імен нових команд та оточень.

Для нумерування ряду об'єктів (розділи, підрозділи, формули, ...) в системі автоматично генеруються відповідні лічильники – створювати нові із таким іменем не можна: при компіляції документу буде виведено повідомлення про помилку.

Приклад створення лічильника із іменем **Num**:

```
\newcounter{Num}
```

Початкове значення створеного лічильника рівне нулю. Для присвоєння йому іншого значення необхідно використати команду (можна використовувати і цілі від'ємні числа):

```
\setcounter{ім'я лічильника}{значення}
```

Наприклад (лічильник присвоюється значення 16):

```
\setcounter{Num}{16}
```

Значення лічильника можна поміняти, додавши до нього задане ціле число (**значення**) за допомогою команди, яка є глобальною не залежно від місця її знаходження:

```
\addtocounter{ім'я лічильника}{значення}
```

Приклад збільшення значення лічильника на одиницю:

```
\addtocounter{Num}{1}
```

Вивести значення лічильника в документі (лічильника **Num**) можна за допомогою команди `{Num}`. Можливі стилі виводу значень приведено нижче:

<code>\arabic{Num}</code>	арабські цифри
<code>\Roman{Num}</code>	римські великі цифри
<code>\roman{Num}</code>	римські малі цифри
<code>\Alph{Num}</code>	нумерація прописними буквами
<code>\alph{Num}</code>	нумерація рядковими буквами

Розглянемо приклад використання лічильника для організації автоматичної нумерації означень у документі. Для вводу означень створимо нове оточення **Oznachennja**, а для їх нумерації використаємо лічильник **Num**. Лічильник збільшує своє значення на 1 при кожному використанні оточення **Oznachennja** командою `\addtocounter{Num}{1}`. Для виводу номерів означень використовуються арабські цифри (`\arabic{Num}`):

```
\newcounter{Num}
\setcounter{Num}{1}
\newenvironment{Oznachennja}[1]{{{ \bf Означення
#1. \Large \addtocounter{Num}{1}}}}
\begin{Oznachennja}{\arabic{Num}}{}%перше означення
Пряма лінія, на якій задано початок відліку,
масштаб та напрям відліку,
називається {\it числовою віссю}.
\end{Oznachennja}
\begin{Oznachennja}{\arabic{Num}}{}%друге означення
{\it Інтервалом} називається множина всіх чисел
(точок),
що знаходяться між будь-якими числами (точками),
які називаються кінцями інтервалу
\end{Oznachennja}
\begin{Oznachennja}{\arabic{Num}}{}%третє означення
Інтервал довжиною $2 \cdot l$ із центром в точці
$a$ називається
```

```

$1-$ о́коло́м то́чки $a$.
\end{Oznachennja}
%четверте означення
\begin{Oznachennja}{\arabic{Num}}{}
{\it Абсолютною величиною} (або {\it модулем })
$\mid {\bf A} \mid$ числа  ${\bf A}$ 
називається число, яке рівне самому  ${\bf A}$ , якщо
 ${\bf A}$  додатне або рівне нулю, і
рівне  $-{\bf A}$ , якщо  ${\bf A}$  від'ємне:
\left[ \mid {\bf A} \mid = \left[ \begin{array}{l} {\bf A} \\ -{\bf A} \end{array} \right. \text{якщо } x \geq 0 \\ -{\bf A} \text{ якщо } x < 0 \end{array} \right]
\end{Oznachennja}
%пя'те означення
\begin{Oznachennja}{\arabic{Num}}{}
Число  ${\bf a}$  називається {\it наближеним}
значенням числа  ${\bf A}$  з {\it похибкою}
 ${\bf \alpha}$ , якщо
$$
{\bf A} - {\bf \alpha} \approx {\bf a}
\end{Oznachennja}

```

Результат компіляції:

Означення 1. Прямá лiнiя, на якiй задано початок вiдлiку, маштаб та напрям вiдлiку, називається *числовою вiссю*.

Означення 2. *Інтервалом* називається множина всiх чисел (точок), що знаходяться мiж будь-якими числами (точками), якi називаються кiнцями iнтервалу

Означення 3. Інтервал довжини $2 \cdot l$ з центром в точцi a називається l -о́коло́м то́чки a .

Означення 4. *Абсолютною величиною* (або *модулем*) $|A|$ числа A називається число, яке рiвне самому A , якщо A додатне або рiвне нулю, i рiвне $-A$, якщо A вiд'ємне:

$$|A| = \begin{cases} A & \text{якщо } x \geq 0, \\ -A & \text{якщо } x < 0. \end{cases}$$

Означення 5. Число a називається *наближеним* значенням числа A з *похибкою* α , якщо

$$A - \alpha \approx a$$

Для нумерації розділів документа використовують лічильники, назви яких відповідають назвам відповідного рівня підрозділів (*chapter, section, subsection* ...). Якщо необхідно, щоби нумерація не було наскрізною по всьому документу, а в кожному розділі своя, можна скористатися відношеннями підлеглості між лічильниками. Для цього лічильник необхідно створити наступним чином:

```
\newcounter{Num} [підрозділ]
```

При кожному збільшенні лічильника відповідного підрозділу лічильник *Num* буде обнуляється автоматично.

Розглянемо приклад створення та використання підлеглих лічильників:

створимо лічильник

```
\newcounter{Num} [section]
```

створимо нове оточення *Oznachennja*:

```
\newenvironment{Oznachennja}[1]{\Large
\addtocounter{Num}{1}} {\bf Означення} \#1. {}
```

Зверніть увагу, що збільшення лічильника на одиницю задано до мітки, яка задає його вивід: щоби не задавати його початкове значення командою `\setcounter{Num}{1}` (початкове значення рівне нулю).

Використаємо в документі створене оточення:

```
\begin{Oznachennja}
{\arabic{section}.\arabic{Num}}
```

```
Прямá лiнiя, на якiй задано початок вiдлiку,
маштаб та напрям вiдлiку,
називається {\it числовою вiссю}.
\end{Oznachennja}
```

В результаті компіляції даного фрагменту отримаємо:

Означення 1.1. Прямá лiнiя, на якiй задано початок вiдлiку, маштаб та напрям вiдлiку, називається *числовою вiссю*.

Перша цифра в нумерації задає номер розділу (*section*), а друга – порядковий номер означення в межах даного розділу (*Num*). При переході до нового розділу лічильник *Num* обнуляється автоматично.

Оточення типу *newtheorem*

Оточення *newtheorem* наперед не визначене, тому може використовуватися для різного типу об'єктів: теорем, визначень, лем, припущень, зауважень і т.д.. Воно самостійно організовує нумерацію введених об'єктів, а для нумерації по підрозділах можна використати відповідні підлегли

лічильники. По замовчуванню при виводі заголовку використовується напівжирний шрифт, а для самого формулювання – курсив. Шрифти можна задати явним чином за допомогою команд форматування тексту. При створенні оточення автоматично створюється і відповідний лічильник, ім'я якого співпадає із його іменем. Можливості використання лічильника аналогічні розглянутим вище.

Оточення типу теорема створюється командою:

`\newtheorem {ім'я оточення}{заголовок}`

Наприклад:

`\newtheorem{Thm}{Теорема}`

При використанні даного оточення в документі:

`\begin{Thm}`

Квадрат гіпотенузи дорівнює сумі квадратів катетів

`$$ a^2+b^2=c^2 $$`

`\end{Thm}`

`\begin{Thm}[Піфагор]`

Квадрат гіпотенузи дорівнює сумі квадратів катетів

`$$ a^2+b^2=c^2 $$`

`\end{Thm}`

Отримаємо:

Теорема 1 Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Теорема 2 (Піфагор) Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Другий приклад (теорема №2) ілюструє можливість використання необов'язкового параметру `[Піфагор]` для виводу додаткової інформації.

Стиль виводу нумерації можна поміняти (після створення оточення) командою:

`\renewcommand{\the ім'я оточення}{стиль}`

Можливі стилі виводу лічильників були розглянуті вище. Приклад зміни стилю виводу нумерації для описаного вище оточення `Thm`:

`\renewcommand{\theThm}{\Alpha{Thm}}`

Отримаємо:

Теорема А Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Теорема В (Піфагор) Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Для використання підлеглих лічильників оточення необхідно задати наступним чином:

`\newtheorem{ім'я оточення}{заголовок}[розділ]`

Наприклад (друга команда задає стиль виводу лічильників):

`\newtheorem{Thm}{Теорема}[section]`

`\renewcommand{\theThm}{\arabic{section}.\alpha{Thm}}`

В результаті попередній фрагмент буде виглядати наступним чином:

Теорема 1.а Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Теорема 1.б (Піфагор) Квадрат гіпотенузи дорівнює сумі квадратів катетів

$$a^2 + b^2 = c^2$$

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте їх та перегляньте результат.
2. Створіть макрос із двома аргументами для вводу прикладів та введіть за його допомогою 2-3 приклади.
3. Створіть нове оточення для вводу теорем. Введіть 5-6 теорем за допомогою створеного оточення та використайте підлегли лічильники для їх нумерації.
4. За допомогою команди `\newtheorem` створіть нове оточення для вводу припущень. Введіть кілька припущень (3-4) та використайте необов'язковий параметр для виводу додаткової інформації.

Контрольні питання

1. Що таке макрос?
2. Наведіть приклад створення нового макросу без аргументів.
3. Яке призначення аргументів у макросі?
4. Як переозначити вже існуючі команди?
5. Як створити нове оточення аргументів?
6. Як переозначити вже існуючі оточення?
7. Що таке лічильник та як його створити?
8. Як задати стиль виводу значення лічильника?
9. Як поміняти початкове значення лічильника?
10. Як збільшити (зменшити значення лічильника на задану величину?
11. Що таке підлеглі лічильники та як їх створити?
12. Перерахуйте стандартні лічильники, що автоматично створюються в **LaTeX**?
13. Яке призначення оточення *newtheorem*?

Лабораторна робота №9

Тема роботи: Графіка у видавничій системі **LaTeX**

Мета роботи: вставка малюнків із графічного файлу та основні операції із боксами. Створення псевдомалюнків.

Для виконання роботи необхідно знати:

- команди включення графічного файлу в документ **LaTeX**;
- можливості псевдографіки системи **LaTeX**;
- побудову графіків засобами **LaTeX**.

Теоретичні відомості

Є дві принципово різні можливості роботи із графікою в документі **LaTeX**. З одного боку - це створити зображення за допомогою спеціалізованих програмних пакетів а потім включити його в документ. З іншого боку - створити її за допомогою базового чи спеціалізованих пакетів **LaTeX**.

Включення графічного файлу

Для включення графічного файлу необхідно в преамбулі підключити пакет **graphicx** та скористатися командою `\includegraphics` в самому документ для вставки малюнку:

`\includegraphics[параметри]{ім'я графічного файлу}`

тут:

параметри задають ряд ключів, які задаються через кому:

width задає розмір малюнка по ширині;

height задає розмір малюнка по висоті;

angle дозволяє повернути малюнок проти годинникової стрілки на даний кут. Від'ємне значення задає поворот за годинниковою стрілкою. Кут задається в градусах, а точка обертання – лівий нижній кут малюнку;

scale масштабує малюнок в цілому. Так, **Scale=0.9** зменшить малюнок на 10 відсотків.

ім'я графічного файлу: ім'я файлу, що вставляється в документ. Можна використовувати ряд графічних форматів: **eps**, **png**, **jpg**. Якщо графічний та вихідний файли знаходяться в різних каталогах, то необхідно задати повне ім'я графічного файлу.

Якщо ширина чи висота малюнку явним чином не задана, то вона обчислюється автоматично виходячи зі збереження масштабу. Положення

малюнку в документі (верхнього лівого кута) визначається місцем виклику команди.

Наприклад:

```
\begin{figure}{Малюнок №1}
\begin{center}
\includegraphics[angle=45,width=0.3\textwidth]
{Graf.jpg}
\end{center}
\end{figure}
```

Зображення спочатку буде повернуто на 45 градусів, а потім змасштабоване до 0.3 ширини сторінки (*textwidth*) зі збереженням масштабу, бо висота явно не задана. Оточення *figure* дозволяє, при необхідності, формувати підпис до малюнку. Ширину та висоту зображення також можна задати явним чином у будь-яких допустимих одиницях вимірювання. Наприклад:

```
includegraphics[width=1.5cm,height=3cm]{Graf.jpg}
```

Команда може використовуватися як для вставки окремих малюнків, так і для вставки невеликих графічних зображень безпосередньо в текст. Наприклад:

Приклад вставки графічного зображення

```
\includegraphics[width=0.3cm,height=0.3cm]{Graf.jpg}
```

безпосередньо в текст документу

Псевдографіка в LaTeX

Створювати малюнки прямо в документі LaTeX дозволяє оточення *picture*. Оточення має значні обмеження на значення параметрів малювання відрізків та діаметрів кіл – вони можуть приймати тільки набір певних значень. Малюнок може бути вставлений у текст документу, або сформований як виключний об'єкт.

Оточення створюється за допомогою команди:

```
\begin{picture}(x,y)
Команди малювання графічних примітивів
\end{picture}
```

(*x,y*) визначають розміри прямокутної області для малюнку. По замовчуванню для задання всіх розмірів використовуються пункти (pt), а початок координат знаходиться в лівому нижньому куті виділеної області.

Команди малювання графічних примітивів

Команди малювання мають дві форми запису:

малювання одиничного об'єкту:

```
\put(x,y){об'єкт}
```

малювання групи об'єктів, що повторюються із певним кроком:

```
\multiput(x,y)(dx,dy){n}{об'єкт}
```

dx, dy – зсув зображення відносно попереднього положення;
n – кількість копій об'єкту.

Основні графічні примітиви (об'єкти)

Текст:

```
\put(x,y){Текст}
```

(*x,y*) – координати початку області вводу тексту.

Відрізок:

```
\put(x,y){\line(x0,y0){Довжина}}
```

x0,y0 – відношення чисел косинус кута нахилу відрізка до горизонталі. Числа не повинні мати спільних дільників (крім одиниці) та можуть приймати значення: –6, –5, ..., 5, 6.

Довжина – довжина проекції відрізка на горизонтальну вісь. Для вертикального відрізка задається його довжина по вертикалі.

Вектор:

```
\put(x,y){\vector(x0,y0){Довжина}}
```

Синтаксис команди аналогічний *line*, але цілі числа, що задають нахил вектора, можуть приймати значення –4, –3, ..., 3, 4.

Коло:

```
\put(x,y){\circle{Діаметр}}
```

Круг:

```
\put(x,y){\circle*{Діаметр}}
```

Точкою відліку для круга та кола є їх центр.

Овал:

```
\put(x,y){\oval(x0,y0){параметри}}
```

x0,y0 – ширина та висота овалу. Точкою відліку для овалу є його центр. *Параметри* дозволяють будувати не повний овал, а тільки його частину:

t – верхня частина овалу;

b – нижня частина овалу;

x – права частина овалу;

1 - ліва частина овалу.

Для малювання декількох частин овалу можна використовувати кілька параметрів одночасно.

Квадратичні криві Безьє

`\qbezier (x,y) (x0,y0) (x1,y1)`

Задається трьома базовими (опорними) точками:

x,y – координати початку кривої;

x0,y0 – координати точки перегину;

x1,y1 – координати кінця кривої;

Товщина ліній

товщина похилих відрізків, кіл та овалів задається командами:

`\thinlines` (тонка лінія)

`\thicklines` (товста лінія)

товщина горизонтальних і вертикальних відрізків та квадратичних кривих Безьє задається командою:

`\linethickness{товщина}`

товщина – задає товщину лінії.

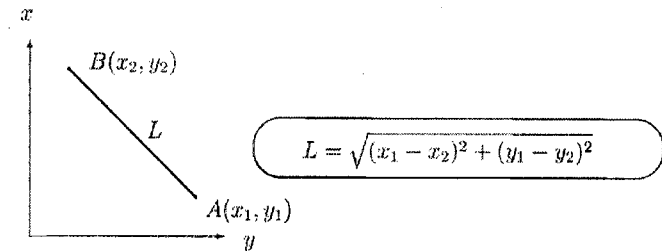
Приклад використання псевдографіки для створення малюнку:

```
\begin{picture} (400,200)
\put(15,150){Відстань між точками на координатній
площині}
\put(20,20){\vector(1,0){100}}
\put(20,20){\vector(0,1){100}}
\put(15,130){$x$}
\put(130,15){$y$}
\put(105,40){\circle*{2}}
\put(110,30){$A(x_1,y_1)$}
\put(40,105){\circle*{2}}
\put(50,105){$B(x_2,y_2)$}
\thicklines
{\put(105,40){\line(-1,1){65}}}
\put(80,70){$L$}
%вставка формули у малюнок
\put(160,60){$L=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$}
\put(240,65){\oval(210,30)}
\end{picture}
\begin{picture} (200,400)
\put(10,5){\line(1,5){30}}
```

`\end{picture}`

Результат компіляції:

Відстань між точками на координатній площині



Побудова графіків в LaTeX

У документ можна вставити графіки, створені безпосередньо засобами LaTeX. Із всіх графічних можливостей розглянемо два пакети:

tikz – пакет для програмування векторної графіки;

pgfplots – доповненням до попереднього пакету для побудови графіків на основі табличних даних.

Для використання базових можливостей пакетів вони повинні бути підключені в преамбулі документу:

`\usepackage{tikz}`

`\usepackage{pgfplots}`

Сам графік формується в межах спеціального оточення **tikzpicture**:

`\begin{tikzpicture}`

Код графіка

`\end{tikzpicture}`

Сам графік (Код графіка) будується за допомогою оточення **axis** (основні параметри оточення зібрано в Додатку №6) та команди `\addplot` (можливі параметри команди винесено в додаток №7). Типи маркерів, що можна використати для побудови графіків приведено в додатку №8. Більш гнучкою у використанні є команда `\addplot+`. Графічні можливості пакетів проілюструємо на ряді прикладів.

Побудова графіка функції по точках.

`\begin{tikzpicture}`

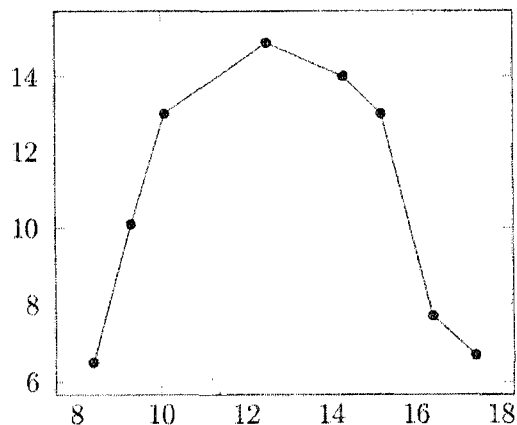
`\begin{axis}`

%задаються координати точок для побудови графіка

%координати точок задаються у форматі: (x,y)

```
\addplot coordinates {
( 8.4, 6.5 )
( 9.3, 10.1 )
( 10.1, 13.0 )
( 12.5, 14.9 )
( 14.3, 14.0 )
( 15.2, 13.0 )
( 16.4, 7.7 )
( 17.4, 6.7 )};
\end{axis}
\end{tikzpicture}
```

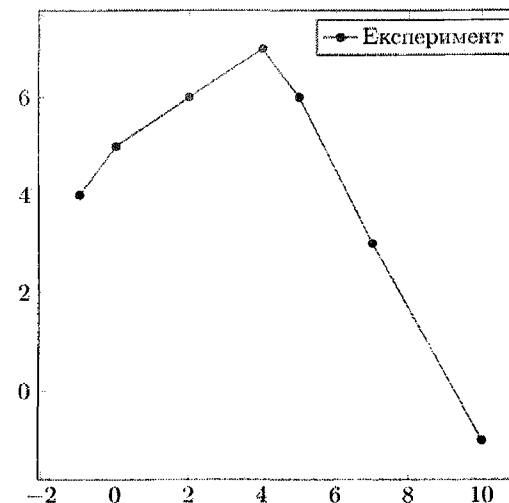
Результат компіляції:



Побудова графіка функції по даних із файлу:

```
begin{tikzpicture}[scale=0.9]
% задано розмір графіка height=10cm,width=10cm
\begin{axis}[height=10cm,width=10cm]
%дані читаються із файлу pts.txt
\addplot file {pts.txt};
% до графіка додається легенда: Експеримент
\addlegendentry{Експеримент}
\end{axis}
\end{tikzpicture}
```

Результат компіляції:



Дані читаються із файлу **pts.txt**, який містить координати точок у два ряди, як це показано на малюнку нижче.

pts.txt	
Файл	Правка
Вид	Настройки
Справка	
%x	y
-1	4
0	5
2	6
4	7
5	6
7	3
10	-1

Файл може бути підготовлений або за допомогою *Блокнота*, або за допомогою будь-якої зовнішньої програми. У файл із даними можна додати необхідні коментарі – рядок, що починається символом **%** або **#** при побудові графіка буде проігнорований. Команда **\addlegendentry{Експеримент}** додає легенду до графіка.

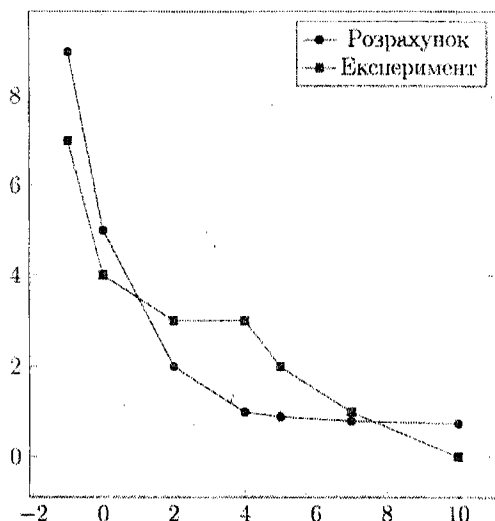
Параметр **scale** оточення **tikzpicture** дозволяє задати масштаб області для побудови. У прикладі вище він складає 90% (**scale=0.9**) від реально заданих розмірів (**height=10cm, width= 10cm**)

Якщо у файлі міститься декілька колонок даних, то команда `\addplot table` дозволяє будувати графіки функцій для вказаних наборів даних. Наприклад:

```
\newpage
```

```
% параметр scale задає масштаб області побудови
\begin{tikzpicture} [scale=0.9]
  \begin{axis} [height=10cm,width=10cm]
    % задання колонок даних для побудови двох графіків
    % дані читаються із файлу pts.txt
    \addplot table [x=x,y=f2(x)] {pts.txt};
    \addplot table [x=x,y=f1(x)] {pts.txt};
    % вивід легенди до першого графіка
    \addlegendentry{Розрахунок}
    % вивід легенди до другого графіка
    \addlegendentry{Експеримент}
  \end{axis}
\end{tikzpicture}
```

Результат компіляції:



Для побудови сформовано файл `pts.txt`. Зверніть увагу, що перший рядок файлу із даними містить назви відповідних стовпців. Назви використовуються для вибору відповідних наборів даних при побудові.

pts.txt - K&L&P&D				
Файл	Правка	Вид	Настройки	Справка
x	f1(x)	f2(x)	f3(x)	f4(x)
-1	4	7	9	
0	5	4	5	
2	6	3	2	
4	7	3	1	
5	6	2	0.9	
7	3	1	0.8	
10	-1	0	0.75	

Побудова кожного із графіків здійснюється командою:

```
\addplot table [x=x,y=f2(x)] {pts.txt};
```

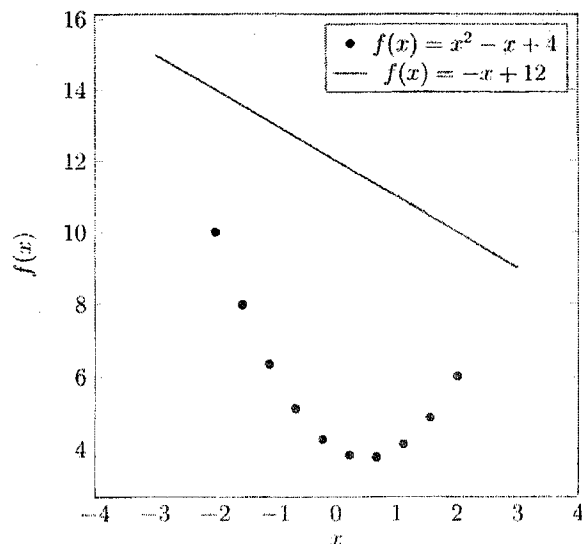
Вибір набору даних, що відображаються на графіку, здійснюється за допомогою параметрів `[x=x,y=f2(x)]` – дані беруться із першого та четвертого стовпчиків файлу. Аналогічним чином будується другий графік із параметрами `[x=x,y=f1(x)]`.

Побудова графіка за заданою функцією

У прикладі будуються графіки двох функцій ($f(x) = x^2 - x + 4$ та $f(x) = -x + 12$) на одній системі координат. Приклад ілюструє використання `LaTeX` як калькулятора. Зверніть увагу на параметри команди `\addplot`, що використані при побудові графіка (див додаток №7):

```
\begin{tikzpicture} [scale=0.9]
  \begin{axis} [xmin=-4,xmax=4,
    xlabel=$x$,
    ylabel={$f(x)$},
    height=10cm,width=10cm]
    % побудова графіка першої функції
    \addplot+[domain=-2:2,draw=blue,samples=10,only
    marks]{x^2 - x + 4};
    % легенда до першого графіка
    \addlegendentry{$f(x)=x^2 - x + 4$}
    % побудова графіка другої функції
    \addplot+[ domain=-3:3,thick,mark=none,draw=red,
    samples=10]{-x + 12};
    % легенда до першого графіка
    \addlegendentry{$f(x)=-x + 12$}
  \end{axis}
\end{tikzpicture}
```

Результат компіляції:



Для побудови використано команду

```
\addplot+[domain=-2:2,draw=blue,samples=10,only
marks]{x^2 - x +4};
```

тут:

$\{x^2 - x + 4\}$ – обов'язків параметр: аналітично задана функція ($f(x) = x^2 - x + 4$), графік якої будується;
 $domain=-2:2$ - діапазон побудови графіка даної функції;
 $samples=10$ - кількість точок, по яких будується графік.

Малюнки та графіка як плаваючі об'єкти

Для управління розміщення малюнків (графіків) в **LaTeX** визначають плаваючий об'єкт за допомогою оточення **figure**. Робота із даним оточенням аналогічна оточенням, що використовуються при створенні іншого типу «плаваючих» об'єктів (див. наприклад ст.109). Якщо необхідно розмістити рисунок безумовно в певному місці (наприклад внизу сторінки), то необхідно використати знаку оклику (**[b!]**).

Приклад використання «плаваючого» об'єкту для оформлення графіка функції:

```
\begin{figure} [ht]
```

```
\centering%розміщення графіка по центру
\begin{tikzpicture}
\begin{axis}
\addplot coordinates {
( 8.4, 6.5 )
( 9.3, 10.1 )
( 10.1, 13.0 )
( 12.5, 14.9 )
( 14.3, 14.0 )
( 15.2, 13.0 )
( 16.4, 7.7 )
( 17.4, 6.7 )};
\end{axis}
\end{tikzpicture}
%формування підпису до графіка
\caption{Графік функції }\label{fig1}
% мітка fig1 може використовуватися для
% організації посилань в межах документу
\end{figure}
```

Результат компіляції:

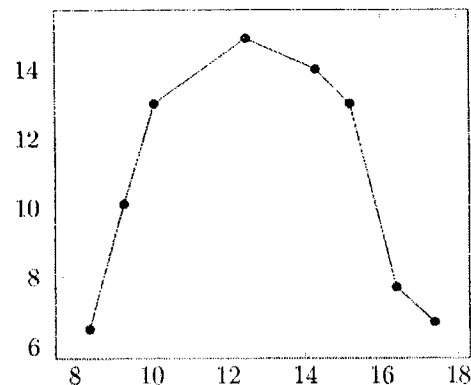
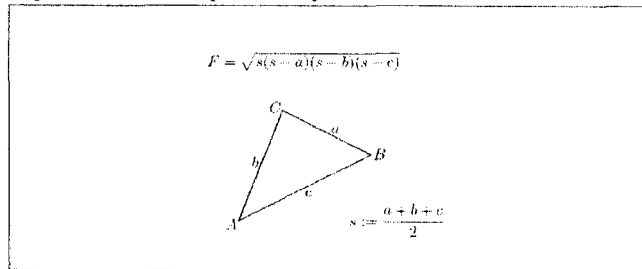


Рис. 1: Графік функції

Хід роботи

- Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте їх та перегляньте результат.
- Відповідно до індивідуального завдання створіть плаваючі об'єкти із підписами:
 - попередньо підготовлений малюнок;
 - графік функції $f_1(x)$ побудуйте по точках (координати точок розрахуйте самостійно) та додайте легенду;
 - графік функції $f_2(x)$ побудуйте, взявши дані із файлу (файл із даними підготуйте самостійно);
 - графік функції $f_3(x)$ побудуйте, використавши **LaTeX** як калькулятор;
 - графіки функцій $f_1(x)$, $f_2(x)$ та $f_3(x)$ побудуйте на одній системі координат (спосіб побудови – на ваш вибір).
- На окремій сторінці, використовуючи команди малювання графічних примітивів, створіть наступний малюнок:



Завдання для індивідуального виконання

1.	Функції	Область побудови та крок
1	2	3
1.	$f_1(x) = x^4 - 8x + 7x = 5$ $f_2(x) = x^4 + 4x^2 - 40x + 3$ $f_3(x) = 5 \cos(2x + \pi)$	$[-1, 2]$, крок $h = 0.25$
2	$f_1(x) = \cos^2(x) + 2x$ $f_2(x) = x^4 - 4x^2 - 20x + 2$ $f_3(x) = -5x + 4$	$[-2, 2]$, крок $h = 0.1$

1	2	3
3	$f_1(x) = \frac{\sin(x^5)}{x}$ $f_2(x) = 3x^5 + 20x^2 - 20x + 3$ $f_3(x) = \ln(x^2)$	$[1, 4]$, крок $h = 0.2$
4	$f_1(x) = x^4 - 4x^3 - 12x + 5$ $f_2(x) = \sin(x - 0.5)$ $f_3(x) = \sqrt[3]{2x - 1}$	$[1, 5]$, крок $h = 0.4$
5	$f_1(x) = \cos(\sqrt{x})$ $f_2(x) = 3x + 4$ $f_3(x) = \ln(4x + 1)$	$[2, 5]$, крок $h = 0.2$
6	$f_1(x) = \left(x + \frac{2}{x}\right)$ $f_2(x) = \ln(x + 4)$ $f_3(x) = 0.1x^3 - x^2 + 2x + 0.4$	$[1, 3]$, крок $h = 0.1$
7	$f_1(x) = \frac{x^2}{x^2 + 1}$ $f_2(x) = -2x^2 + \frac{1}{3x}$ $f_3(x) = \cos\left(x + \frac{1}{x}\right)$	$[1.3, 5.1]$, крок $h = 0.3$
8	$f_1(x) = x^3 - 0.3x^2 - 2.9$ $f_2(x) = \cos\left(\frac{x}{x + 1.7}\right)^2$ $f_3(x) = \frac{x}{x^2 - 2.5}$	$[3, 7]$, крок $h = 0.15$
9	$f_1(x) = \frac{1}{\sqrt{2x + 3}}$ $f_2(x) = 0.3x^4 - 2x^3 + 3x$ $f_3(x) = x^3 \cdot \cos(x)^2$	$[-1, 1]$, крок $h = 0.1$

1	2	3
10	$f_1(x) = \sin\left(\frac{x^2}{2}\right)$ $f_2(x) = \left(\frac{x^2}{1+x}\right)$ $f(x) = (x+1)^3 - x$	[1,4], шаг $h = 0.2$
11	$f_1(x) = e^x + 2$ $f_2(x) = 3 - x^2$ $f_3(x) = 3x^4 - 12x^2 - 60x + 2$	[-1,1], шаг $h = 0.05$
12	$f_1(x) = \cos(x) - 10 \cdot x + 1$ $f_2(x) = x^3 - x - 2$ $f_3(x) = \frac{1}{4+x^2}$	[-3,3], шаг $h = 0.5$
13	$f_1(x) = x^3 + x - 1$ $f_2(x) = x - \sin(x) + 1$ $f_3(x) = e^x - x - 1$	[-2,2], шаг $h = 0.2$
14	$f_1(x) = x^4 - 2x^2 - 4x + 5$ $f_2(x) = x - \frac{1}{(1+x)^3}$ $f_3(x) = x - 2x \sin(0.5x)$	[0,2], шаг $h = 0.1$
15	$f_1(x) = x + 3 \ln(x)$ $f_2(x) = \lg(x) + x$ $f_3(x) = 3 \cdot \ln\left(1 + \frac{x}{2}\right) - 2$	[2,8], шаг $h = 0.5$
16	$f_1(x) = x^4 - 8x^2 + 8x + 5$ $f_2(x) = 3 \ln\left(1 + \frac{x}{2}\right) - 2$ $f_3(x) = 10x - 4 \cdot \sin(0.5x)$	[0,3], шаг $h = 0.15$

1	2	3
17	$f_1(x) = \cos(x) - \sqrt{x}$ $f_2(x) = 6x - 2$ $f_3(x) = \cos(0.5x) + 3$	[1,5], шаг $h = 0.4$
18	$f_1(x) = 3 \ln(3+x) - 2$ $f_2(x) = x^2 + 5x$ $f_3(x) = \frac{x}{e^x}$	[-1,2], шаг $h = 0.2$
19	$f_1(x) = x^4 - 4x^3 - 12x + 1$ $f_2(x) = e^{\frac{x}{2}}$ $f_3(x) = \sqrt{\frac{1+x^2}{5-x^2}}$	[-2,2], шаг $h = 0.25$
20	$f_1(x) = x^4 + 8x^3 - 12x + 5$ $f_2(x) = \sqrt{1 + \ln(x)^2}$ $f_3(x) = \cos(x^2 + 2) - 1$	[3,10], шаг $h = 0.5$
21	$f_1(x) = \ln(x)^2$ $f_2(x) = \sqrt{1+x^2}$ $f_3(x) = x \cos(5x)$	[2,4], шаг $h = 0.1$
22	$f_1(x) = x^3 - 0.3x^2 - 3x$ $f_2(x) = \sin(3x)$ $f_3(x) = x + \sqrt{x}$	[0,2], шаг $h = 0.2$
23	$f_1(x) = x - \sqrt{x}$ $f_2(x) = \sin(3x)$ $f_3(x) = 3x^4 - 16x^3 + 24x$	[0,1], шаг $h = 0.05$
24	$f_1(x) = -x^3 + 0.3x^2 - 3.1x - 1$ $f_2(x) = x^2 - \sqrt{x}$ $f_3(x) = \sin^2\left(\frac{x}{2}\right)$	[0,2], шаг $h = 0.15$

1	2	3
25	$f_1(x) = \cos^2(3x)$ $f_2(x) = \sqrt{(x^2 + 4)}$ $f_3(x) = 2 - \sin(2x)$	$[-3,3]$, крок $h = 0.5$
26	$f_1(x) = 3x^2 - 4x + 1$ $f_2(x) = \frac{\sin(3x)}{x}$ $f_3(x) = \lg(3x) + 2x$	$[0.1,4.2]$, крок $h = 0.2$
27	$f_1(x) = x^6 - 12x + 18$ $f_2(x) = \sin\left(\frac{x^2}{3}\right)$ $f_3(x) = \sqrt{3 - x^2}$	$[-1,1]$, крок $h = 0.1$
28	$f_1(x) = x^4 + 2x^2 - 20x + 3$ $f_2(x) = \frac{x+3}{x-2}$ $f_3(x) = \sin(7x + 3)$	$[-2,1]$, крок $h = 0.25$
29	$f_1(x) = \ln(2 + x)$ $f_2(x) = \frac{x^2}{5 + x}$ $f_3(x) = \frac{x}{\cos^2(x)}$	$[-0.5,0.5]$, крок $h = 0.05$
30	$f_1(x) = 1.3x^3 + 6.4x^2 - 1.5x - 27$ $f_2(x) = 2x + 3\cos(x)$ $f_3(x) = \cos(3x + \pi) + 2x^2$	$[-1.5,1.5]$, крок $h = 0.15$

Контрольні питання

1. Як вставити малюнок в документ?
2. Яке оточення служить для створення малюнку засобами **LaTeX**?
3. Перерахуйте основні графічні примітиви для створення малюнків.
4. Як створити коло?
5. Як задати товщину лінії?
6. Як побудувати графік засобами **LaTeX**?
7. Як побудувати графік функції за даними із файлу?
8. Перерахуйте основні параметри оточення **axis**.
9. Які параметри команди **addplot** ви знаєте?
10. Як задати тип маркера, що буде використовуватися при побудові графіка?
11. Як до графіку додати легенду?
12. Як побудувати кілька графіків на одній системі координат?
13. Як створити малюнок як «плаваючий» об'єкт?
14. Як створити підпис до графіка?
15. Для чого служить мітка при створенні «плаваючого» об'єкту?

Лабораторна робота №10

Тема роботи: Створення комп'ютерних презентацій у видавничій системі **LaTeX**

Мета роботи: ознайомлення із основними можливостями та порядком створення електронних презентацій в пакеті **beamer**

Для виконання роботи необхідно знати:

- загальну структуру **LaTeX** - документу для створення електронної презентації;
- основні теми пакету;
- команди формування фрейму презентації;
- роботу із кольором та фоном презентації;
- інтерактивні елементи презентації;
- створення динамічних фреймів.

Теоретичні відомості

Одним із найбільш розвинутих та найпопулярніших для створення електронних презентацій є пакет **beamer**. Він дозволяє:

- використовувати стандартні команди системи **LaTeX**;
- створювати елементи, що перекриваються;
- створювати динамічні об'єкти;
- використовувати стандартні шаблони оформлення презентацій.

В класі **beamer** презентація складається із набору окремих слайдів (фреймів), кожен із яких може мати кілька сторінок (шарів, які використовуються для організації динамічних ефектів при показі). Для того, щоби створити презентацію необхідно створити відповідний **tex** – документ, скомпілювати його та отримати **pdf** - файл. Для запуску презентації на перегляд необхідно відкрити її за допомогою **Acrobat Readera** та включити повноекранний режим.

Розмір віртуальної сторінки в **beamer** рівний **128x96mm**, а вікно фрейму складається із ряду елементів, присутність яких на сторінці визначається темою оформлення:

- верхня та нижня панелі;
- ліва та права бокові панелі;
- панелі навігації;
- навігаційні значки;
- малюнок-логотип;
- заголовок слайду;

- фон слайду;
- основне наповнення слайду.

Зовнішній вигляд презентації задається за допомогою п'яти груп тем (із повним переліком доступних тем можна ознайомитись, переглянувши вміст директорії **texmf\text\latex\beamer\base\themes**):

- **основна тема (theme)**: задає зовнішній вигляд всіх базових елементів презентації. Задається в преамбулі командою **\usetheme[опції]{тема}**. До складу пакету входять наступні теми: **AnnArbor, Antibes, Bergen, Berkeley, Berlin, Boadilla, boxes, CambridgeUS, Copenhagen, Darmstadt, default, Dresden, Frankfurt, Goettingen, ...**;
- **внутрішня тема (inner)**: визначає вигляд елементів в межах робочої області презентації. Задається в преамбулі командою **\useinnertheme[опції]{тема}**. До складу пакету входять теми: **default, circles, rectangles, rounded, inmargin**;
- **зовнішня тема (outer)**: визначає загальну розмітку слайду. Задається в преамбулі документу командою **\useoutertheme[опції]{тема}**. До пакету входять теми: **default, infolines, miniframes, smoothbars, sidebar, split, shadow, tree, smoothtree**;
- **колірна схема (color)**: визначає колір основних елементів презентації. Задається в преамбулі командою **\usecolortheme[опції]{тема}**. Із повним переліком кольорових схем, визначеними в пакеті, можна ознайомитись, переглянувши вміст директорії **texmf\text\latex\beamer\base\themes\color**;
- **шрифтова схема (font)**: визначає шрифт основних елементів презентації. Задається в преамбулі командою **\usefonttheme[опції]{тема}**. До складу пакету входять наступні теми: **default, serif, _structurebold, structureitalicserif, structuresmallcapsserif**.

Розмір шрифту в межах всього документу задається як параметру класу **beamer** аналогічно до звичайного документу **LaTeX**. Підтримуються розміри шрифтів: **8pt, 9pt, 10pt, 11pt, 12pt, 14pt, 17pt** та **20pt**. Наприклад:

```
\documentclass[10pt]{beamer}
```

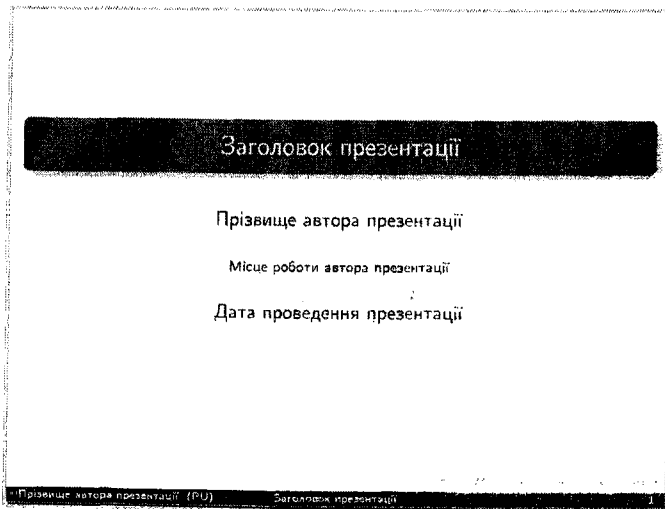
задає базовий розмір шрифту **10pt**, а всі інші розміри в презентації розраховуються виходячи із нього.

Загальна структура документу

Загальна структура документу для створення презентації є наступною:

```
\documentclass{beamer}
\usepackage[cp1251]{inputenc}
\usepackage[ukrainian]{babel}
%задання основної теми
\usetheme {Madrid}
\title {Заголовок презентації}
\date{Дата проведення презентації}
\author{Прізвище автора презентації}
\institute[PU]{Місце роботи автора презентації}
\begin{document}
%перший фрейм презентації
\begin{frame} [t]
%формування заголовка презентації
\titlepage
\end{frame}
\end{document}
```

Результат компіляції:



Кожен фрейм презентації оформляється за допомогою оточення **frame**. Все, що знаходиться між **\begin{frame}[параметр]** та **\end{frame}** розміщується на окремому слайді. **Параметр** (може бути опущеним) управляє розміщенням тексту на слайді:

- t** - текст «притискається» по верхнього краю слайду;
- b** - текст «притискається» до нижнього краю слайду;
- c** - текст вирівнюється по центру слайду.

У прикладі вище на першому слайді формується титульна сторінка презентації за допомогою команди **\titlepage**. Необов'язковий параметр до команд **\author**, **\title**, **\date**, **\institute** дозволяє задати короткий текст, що буде виводитись на інформаційній панелі слайду. Наприклад:

```
\institute[Університет]{Місце роботи автора презентації}
```

Додати логотип (зображення, що буде відображатися на всіх слайдах) до презентації можна командою **logo**. Наприклад:

```
\logo{\includegraphics[height=0.5cm]{Graf.jpg}}
```

тут

height=0.5cm – розмір логотипу;

Graf.jpg – файлу із зображенням, яке буде використовуватися в якості логотипу.

Інформаційна панель розміщується внизу кожного слайду та може бути включена за допомогою команди (якщо це не передбачено використовуваною темою):

```
\useoutertheme{infoclines}
```

Команди, що використовуються при формуванні слайдів

Заголовки до фреймів формують командою **\frametitle{Заголовок слайду}**, або як параметр оточення **frame**. До фрейму може також додати підзаголовок за допомогою команди **\framesubtitle{Підзаголовок слайду}**. Наприклад:

```
\begin{frame} [t] {Заголовок слайду}
```

Для структурування презентації за межами оточення **frame** можуть використовуватися розділи **\section{Назва розділу}** та підрозділи **\subsection{Назва підрозділу}**. Вони полегшують швидкий доступ до слайдів відповідних розділів та дозволяють командою **\tableofcontents** сформувати зміст презентації.

Оверлей – фрейм, що складається із кількох сторінок, які накладаються одна на одну. Для його створення використовується команда `\pause`. Команда приводить до створення двох сторінок фрейму: на першій виводиться текст слайду до команди, а на другому додатково буде виведено і текст після команди (до наступної команди `\pause`, якщо вона присутня). Приклад створення фрейму, що складається із трьох сторінок:

```
\begin{frame}[t]
\begin{itemize}
\item{Перша сторінка}
\pause
\item{Друга сторінка}
\pause
\item{Третя сторінка}
\end{itemize}
\end{frame}
```

Команди в межах фрейму можуть використовувати специфікації перекриття сторінок: в кутових дужках після команди задається перелік номерів чи діапазонів сторінок фрейму, на яких діє команда чи відображається текст. Наприклад формування маркованого списку із використанням специфікації перекриття сторінок:

```
\begin{itemize}
\begin{frame}[t]
\item<1-3>{\textbf{Текст присутній на всіх
сторінках
(на першій він виділений жирним)}}
\pause
\item<2>{Текст присутній тільки на другій
сторінці}
\pause
\item<3>{Текст присутній тільки на третій
сторінці}
\end{itemize}
\end{frame}
```

Для організації динамічної зміни елементів фрейму, що розміщені в певному місці слайду, також можна скористатися оточенням **overprint**, як це проілюстровано на прикладі нижче (фрейм складається із двох сторінок, кожна із яких створюється командою `\onslide`, і для якої теж задається специфікація перекриття):

```
\begin{frame}[t]
%формування титульної сторінки презентації
\begin{overprint}
\onslide<1>
Перша сторінка слайду
\onslide<2>
Друга сторінка слайду
\end{overprint}
\end{frame}
```

Робота з кольором

Для роботи із кольором в преамбулі документу необхідно підключити пакет `\usepackage{xcolor}`. Колір тексту задається командою:

```
\textcolor{колір}{Текст буде виділено заданим
кольором}
```

тут:

колір – колір тексту англійською мовою (**red, orange, yellow, green, cyan, blue, violet**).

Будь-який власний колір можна визначити наступною командою в преамбулі документу чи в самому документі за межами оточення **frame** (при вводі команди в межах оточення **frame** він визначений тільки в його межах):

```
\definecolor{ім'я кольору}{колірна модель}{черво-
ний, зелений, синій}
```

тут:

ім'я кольору – ім'я, що буде використовуватися для доступу до даного кольору;

колірна модель – використовуватимемо **RGB** колірну модель, в якій будь-який колір формується як суміш трьох базових кольорів: **червоного, зеленого та синього**.

{червоний, зелений, синій} – інтенсивність базових кольорів в межах від 0 (мінімальна інтенсивність - колір виключено) до 1 (максимальна інтенсивність). Наприклад:

```
%визначення власного кольору coll
\definecolor{coll}{rgb}{0.1,0.4,0.2}
%застосування власного кольору coll до тексту
\textcolor{coll}{Для даного тексту
використано колір coll}
```

Виділити тексту кольоровою рамкою із фоном даного кольору можна за допомогою команди:

```
\fcolorbox{колір рамки}{колір фону}{текст}
```

Наприклад:

```
\fcolorbox{red}{green}{Текст буде взято в червону  
рамку із зеленим фоном}
```

Робота із блоками

Для виділення фрагментів слайду зручно використовувати блочні конструкції, які можна сформувати з допомогою оточення **block**:

```
\begin{block}{Заголовок блоку}  
Наповнення блоку  
\end{block}
```

Для виділення більш важливої інформації служить оточення **alertblock** (від попереднього воно відрізняється тільки кольоровою гамою):

```
\begin{alertblock}{Заголовок блоку}  
Наповнення блоку  
\end{alertblock}
```

Кольорову гаму блоку можна задати в преамбулі документу явним чином за допомогою наступних команд:

колір заголовку блоку

```
\setbeamercolor{block title}{fg=колір1,bg=колір2}
```

тут

```
Fg=колір1 - колір шрифту заголовку;  
bg=колір2 задає колір фону для заголовку;
```

колір фону

```
\setbeamercolor{block body}{bg=колір}.
```

Робота із фоном фрейму

Колір фону презентації можна міняти за допомогою команди:

```
\beamertemplateshadingbackground{колір1!яскравість  
1}{колір2!яскравість2}
```

тут:

колір1 міняє свою інтенсивність (у відсотках) від значення *яскравість1* внизу і до 0 у верхній частині слайду;

колір2 міняє свою інтенсивність від значення *яскравість2* вгорі слайду і до 0 внизу.

Наприклад:

```
\beamertemplateshadingbackground{green!65}{red!42}
```

В якості фону можна використати будь-який малюнок. Для того, щоби він займав всю область слайду, задано ширину рівною ширині слайду (*\paperwidth*). Наприклад (команда повинна стояти перед тим оточенням **frame**, фон для якого вона задає):

```
\usebackgroundtemplate{  
  {\includegraphics[width=\paperwidth]  
  {Graf.jpg}}  
}
```

Інтерактивні елементи управління презентацією

Гіперпосилання

За роботу із гіперпосиланнями відповідає пакет **hyperref**, який повинен бути підключений в преамбулі документу. Посилання можуть бути як на елементи в середині самої презентації, так і на зовнішні документи чи сторінки мережі **Internet**. Для організації переходів в межах презентації необхідно:

- командою **\label** присвоїти унікальну мітку сторінці фрейму (мітку можна також задати як параметр до оточення **frame**). Міткою можуть бути тільки будь-які комбінації англійських букв, цифр та символів-розділювачів;
- командою **\hyperlink<слайди>{мітка}{текст}** організувати перехід до мітки. Параметр **<слайди>** задає сторінки фрейму, на яких відображається **текст**, що буде гіперпосиланням.

Альтернативною до **\label** є команда **\hypertarget <слайд>{мітка}{текст}**. Вона дозволяє задати мітки для певних сторінок в межах фрейму. **Текст** є гіперпосиланням на слайд, вказаному в параметрі **слайд**. Він може бути опущеним. Команди, які не вимагають використання міток для організації переходів, приведено в додатку №9.

Управляючі кнопки

Команда **\beamerbutton{текст}** дозволяє малювати управляючі кнопки. **Текст** задає напис, що буде розміщено на кнопці. Для налаштування роботи кнопки вона використовується в середині команди створення гіперпосилання. Наприклад:

```
\hyperlink {a1}{\beamerbutton{На початок}}
```

З інші можливостями малювання управляючих кнопок можна ознайомитись в Додатку №10.

Масштабування частини слайду

Окремі області слайду можна показати у збільшеному масштабі. При створенні такої області вона буде гіперпосиланням, активізація якого приведе до її відображення на весь екран на іншій сторінці слайду. Клік миші на збільшеній області повертає презентацію до попередньої сторінки. Для масштабування частини слайду служить команда:

```
\framezoom<без масштабу><у збільшеному масштабі>
[опції] (x,y) (ширина, висота)
```

тут:

<без масштабу> - номер сторінки слайду, на якій буде сформована область для масштабування;

<у збільшеному масштабі> - номер сторінки слайду, на якій буде розгорнуто область на всю сторінку;

[опції] – значення **border** малює рамку навколо області масштабування;

(x,y) – координати верхнього лівого кута області масштабування (відлік координат ведеться від лівого верхнього кута області слайду);

(ширина, висота) – розмір області масштабування.

На одній сторінці слайду можна задати декілька областей для масштабування. Наприклад (область масштабування формується на першій сторінці, а відкривається на другій сторінці; область обведена рамкою):

```
\begin{frame} [t]
```

```
\framezoom<1><2>[border] (0.1cm, 0.1cm) (2cm, 2cm)
```

Прикла слайду, що ілюструє можливість створення слайду із областю масштабування. Область для масштабування буде означена рамкою.

```
\end{frame}
```

Управління переходами між слайдами

Для управління способами переключення (анімаційними ефектами переходів) між слайдами служить приведений в додатку №11 набір команд. Наприклад, команда

```
\transblindshorizontal<1-3>[duration=4]
```

задає зміну 1, 2 та 3 сторінки фрейму горизонтальними полосами. Заміна сторінок проходить за 4 секунди. Ефекти переходу між сторінками працюють тільки при повноекранному режимі перегляду презентації.

Хід роботи

1. Ознайомтесь зі теоретичним матеріалом до роботи. Введіть приклади, скопіюйте та перегляньте результат.
2. Відповідно до індивідуального завдання створіть презентацію на задану тему. При створенні презентації використовуйте всі базові можливості пакету **beamer**.

Завдання для індивідуального виконання

Варіант	Тема презентації
1	2
1.	Системи числення.
2.	Раціональні та ірраціональні числа.
3.	Задачі апроксимації функцій.
4.	Афінні перетворення на площині.
5.	Теорема Ферма.
6.	Геометричні фігури на площині та їх площі.
7.	Історія розвитку математики.
8.	Неевклідова геометрія.
9.	Основні поняття теорії імовірності.
10.	Ітераційні методи пошуку розв'язків.
11.	Випадкові події та їх імовірності.
12.	Основні поняття теорії імовірності.
13.	Елементарні функції.
14.	Теорема Піфагора та методи її доведення.
15.	Тригонометричні функції.
16.	Похідна функції.
17.	Чисельні методи розв'язку рівнянь.
18.	Методи інтегрування.
19.	Чисельне інтегрування функцій.
20.	Дослідження функції за допомогою похідної.
21.	Метод математичної індукції.
22.	Системи координат.
23.	Видатні древньогрецькі математики.
24.	Логарифмічна функція: її властивості та графік.

1	2
25.	Розклад числа на прості множники.
26.	Похідна та її застосування.
27.	Методи інтегрування функцій.
28.	Видатні математики України.
29.	Використання графічного способу при розв'язку математичних задач.
30.	Наближене обчислення означених інтегралів.

Контрольні питання

1. Із яких елементів може складатися вікно фрейму електронної презентації?
2. Перерахуйте групи тем, що задають зовнішній вигляд презентації.
3. Які розміри шрифтів підтримуються в пакеті *beamer*.
4. Яка структура документу в пакеті *beamer*?
5. Як створити заголовок слайду?
6. Для чого використовуються розділи та підрозділи при створенні презентації?
7. Що таке специфікація перекриття сторінок?
8. Як задати колір для тексту та фону?
9. Як визначити власний колір?
10. Що таке блок та як управляти його кольоровою гамою?
11. Перерахуйте команди для роботи із фоном презентації?
12. Як додати в презентацію гіперпосилання?
13. Яка команда дозволяє створити управляючі кнопки?
14. Як вивести фрагмент слайду у збільшеному масштабі?
15. Як задати режим переключення між слайдами?

Додаток №1. Повідомлення про помилки транслятора LaTeX (типові помилки, що зустрічаються найчастіше)

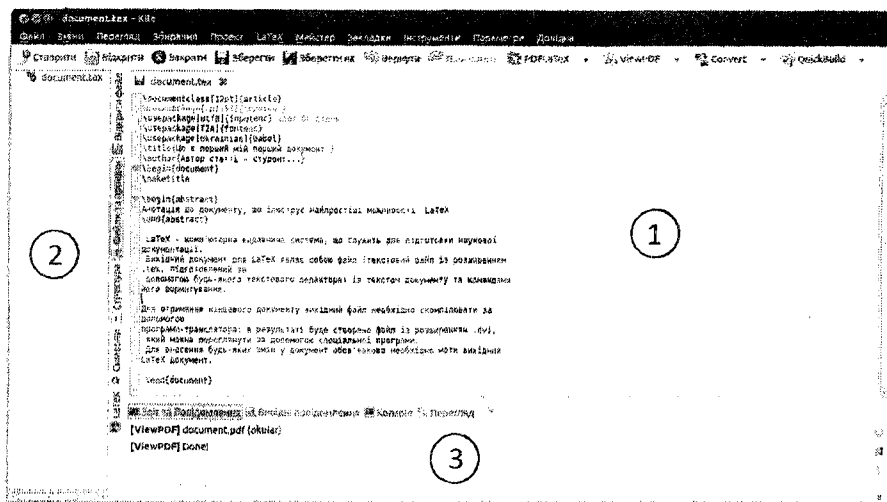
№ п/п	Назва помилки	Опис помилки та способи її виправлення
1	3	3
1.	! LaTeX Error: missing \begin{document}	Текст був поміщений перед <code>\begin{document}</code> або <code>\begin{document}</code> відсутній.
2.	! LaTeX Error: Tab overflow	Занадто багато <code>\=</code> в засобах табуляції.
3.	! LaTeX Error: Environment document undefined	Не правильна назва оточення
4.	! LaTeX Error: \usepackage before \documentclass	Використання пакету перед класом документа Напишіть спочатку клас документа.
5.	! LaTeX Error: File ended while scanning use of \end	Команда не завершена.
6.	LaTeX Error: Extra alignment tab has been changed to \cr	Відсутнє посилання. Необхідно його додати.
7.	! LaTeX Error: Too deeply nested	Помилка пов'язана із неправильним формуванням вкладення одного списку в інший.
8.	! LaTeX Error: Something's wrong--perhaps a missing \item	Швидше за все пропущений аргумент в командних дужках <i>thebibliography</i> .
9.	! LaTeX Error: Runaway argument?	Кількість дужок та їх розташування не відповідає правилам оформлення в LaTeX
10.	! LaTeX Error: Suggested extra height (14454.0pt) dangerously large.	В <code>\enlargethispage</code> вказане надто велике число
11.	LaTeX Error: Illegal character in array arg	Недопустимий символ в аргументі масиву
12.	\begin{...} ended by \end{...})	LaTeX знайшов тільки кінець чи тільки початок командних дужок. Можлива причина: помилка в імені, пропуск або зайва вставка <code>\begin</code> або <code>\end</code> командних дужок (оточень).

1	2	3
13.	! Too many }'s.	Відкрита фігурна дужка не закрита або закрита не згідно правил.
14.	Environment...undefined	Використовується оточення з невідомим ім'ям.
15.	Too deeply nested	В документі більш ніж чотири вкладених списки з автоматичним переобчисленням.
	Bad use of \\	Команда \\ використовується між абзацами, де вона не має сенсу.
	Missing \begin {document}	Пропущена команда <code>\begin {document}</code> , або в преамбулі міститься текст чи команда, що форматує текст.
	Can be used only in preamble	Команда LaTeX , яка може використовуватися тільки в преамбулі документу, використана після команди <code>\begin {document}</code> . До команд преамбули відносяться: <code>\documentstyle, \noindent, \includeonly, \makeindex i \makeglossary.</code> Таке-ж повідомлення виводиться при вводі зайвої команди <code>\begin {document}</code> .
	Bad math environment delimiter	Невідповідність кількості дужок, що відкривають та закривають математичну формулу.
	Something's wrong--perhaps a missing \item	Скоріш за все пропущено команду <code>\item</code> в оточенні, що використовується для створення списків. Таке-ж повідомлення виводиться, якщо пропущено аргумент оточення <i>thebibliography</i> .
	Text line contains an invalid character	Рядок містить заборонені символи. Швидше за все текст має неправильне кодування.
	Undefined tab position	Спроба переміщення табулятора на невизначену позицію при використанні однієї з команд: <code>\>, \+, \- або \<</code> .
	Missing {	Аргументи в команді мають непарні фігурні дужки.

1	2	3
23.	Missing number, treated as zero	Не задано числове значення параметру. Швидше за все пропущено обов'язковий аргумент команди, або квадратні дужки розташовані так, що були прийняті за початок необов'язкового аргументу.
24.	\< in mid line	В тілі оточення tabbing використана команда <code>\<</code> в середині рядка. Ця команда може з'являтися тільки на початку рядка.
25.	\begin{...}) ended by \end{...})	Знайдено тільки кінець чи тільки початок командних дужок. Можлива причина: помилка в імені, лишні пробіли або зайва вставка <code>\begin</code> або <code>\end</code> .
26.	Command name name invalid in math mode.	Спроба використати в математичному режимі команду, яка призначена для використання тільки в тексті.

Додаток №2. Редактор Kile

Для роботи із видавничою системою **LaTeX** під управлінням операційної системи **Linux** можна використовувати текстові редактори із відкритим кодом, одним із яких є, наприклад, **Kile**. **Kile** є комплексним середовищем, яке дозволяє скористатися всіма функціональними можливостями **LaTeX** за допомогою зручного графічного інтерфейсу. Він передбачає автоматичне завершення коду в межах документу, його компіляцію та обробку помилок, інструменти перетворення форматів документу та їх перегляду, допоміжні додатки – майстри для формування основних типів документів та інструменти для керування створеними проектами – документами. Елементи коду різного типу підсвічуються різним кольором: команди **LaTeX** будуть відрізнятися від звичайного тексту; математичні формули також будуть підсвічені іншим кольором.



Вікно редактора Kile:

- 1 – робоча область для вводу документу;
- 2 – бічна панель;
- 3 – панель «Журнал і повідомлення».

В панелі «Журнал і повідомлення» виводиться повідомлення про хід роботи з документом: при компіляції тут виводяться повідомлення про помилки у вихідному документі:

```
[LaTeX] document.tex => document.dvi (latex)
[LaTeX] завершено з кодом виходу 1
./document.tex:29:Undefined control sequence \Vfk
[LaTeX] 1 помилка, 0 попереджень, 0 перевищень
```

У повідомленні завжди зазначається номер рядка документу, обробка якого призвела до помилки. Для перегляду рядка, у якому знайдено помилку, необхідно натиснути на повідомленні про помилку на панелі перегляду журналу компіляції: програма встановить курсор на рядку, що містить помилку. Перейти від одного повідомлення до іншого можна за допомогою пунктів меню **Збирання: Попередня помилка, Наступна помилка, Попереднє попередження, Наступне попередження**.

У бічній панелі (див. малюнок нижче), в залежності від вибору пункту, може виводитись наступна інформація:

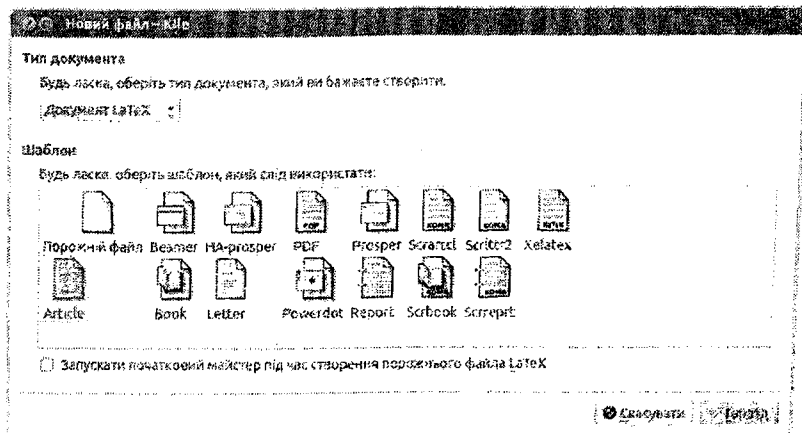
команди LaTeX — список команд та параметрів, які можна їм передати: їх можна вставити до документу за допомогою лівої кнопки миші;

символи — команди вставки спеціальних символів, згрупованих по категоріях;

файли та проекти — структура створеного проекту.



Документ в редакторі можна створити за допомогою Майстра швидкого створення документів, або на основі порожнього. При використанні майстра у меню програми можна вибрати один з багатьох варіантів створення документу та задати ряд параметрів, пов'язаних із його типом:



Створення нового файлу (LaTeX- документу) в Kile

Попередньо визначеними шаблонами у Kile є:

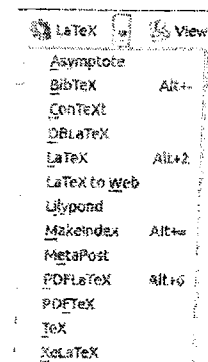
- **Порожній документ (Empty File)** введення починається із «порожнього листа»;
- **Article:** стаття;
- **Report:** звіт;
- **Book:** книга;
- **Letter:** лист;
- **Beamer:** презентація;
- **Powerdot:** презентація у форматі *.pdf*;
- **Scrartcl, Scrbook, Scrrpt, Scrtr2:** класи документів, спеціально адаптовані до німецьких типографських стандартів;
- **Xelatex:** змінений шаблон *article* для використання з XeLaTeX.

Інструменти для збирання документу

Компіляція, перетворення та перегляд

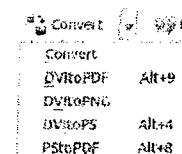
Для перегляду документу його необхідно скомпілювати (зібрати). Всі інструменти збирання згруповано в меню **Збирання-Компілювати**; **Збирання-Перетворити** та **Збирання – Переглянути**.

Компіляція документу:



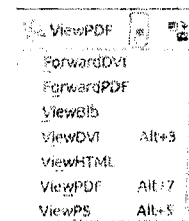
- **LaTeX:** запускає компіляцію активного документу та формує файл із розширенням *.dvi*;
- **PDFLaTeX:** запускає компілятор **PDFLaTeX** для поточного документу та формує файл із розширенням *.pdf*.

Перетворення форматів документу:



- **DVItPDF** перетворює файл формату *.dvi* на *pdf*-файл;
- **DVItPS** перетворює файл формату *.dvi* на *ps*-файл;
- **PStoDVI** перетворює файл формату *.ps* на *pdf*-файл.

Перегляд документів:



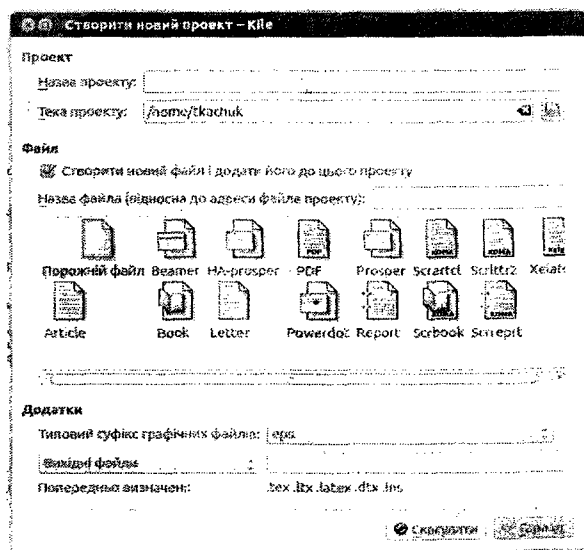
- **ViewDVI:** перегляду файлу формату *.dvi*;
- **ViewPDF:** перегляду файлу формату *.pdf*;
- **ViewPS:** перегляду файлу формату *.ps*;
- **ForwardDVI:** відкриває сторінку файлу формату *.dvi*, що відповідає поточному рядку документу, відкритому у редакторі;
- **ForwardPDF:** відкриває сторінку файлу формату *.pdf*, що відповідає поточному рядку документу, відкритому у редакторі.

Порядок створення документу

1. Запустити редактор.
2. Скористатися пунктом меню **Майстер – Швидкий запуск** та вибрати стиль документу та його параметри.
3. По завершенню роботи майстра ввести текст та відформатувати його за допомогою відповідних команд.
4. Зберегти створений файл, задавши йому назву.
5. Зібрати (відкомпілювати) документ, натиснувши відповідну кнопку **LaTeX**.
6. Якщо зібрати документ в результаті помилок не вдалося, то виправити їх. Повідомлення про помилки можна побачити у області перегляду файлу журналу та повідомлень.
7. Натиснути кнопку **ViewDVI** для перегляду створеного документу;
8. Для повернення до режиму редагування документу натисніть кнопку «**Вікно редактора**» або закрийте вікно перегляду.

Порядок створення проекту

Проект – це група документів, графічних та інших файлів, що містять дані, потрібні для збирання документу. Типовим проектом є документ, що складається з кількох глав, збережених в різних документах. Щоби полегшити керування проектом, всі документи об'єднуються одним проектом, специфікація якого зберігається в окремому документі з розширенням *.kilepr*. Щоби створити проект необхідно скористатися пунктом меню **Проект – Новий проект...**:



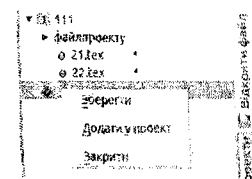
При створенні нового проекту необхідно задати:

- назву проекту;
- назву файлу проекту із розширенням *.kilepr*.
- назву файлу документу;
- тип файлу, який буде створено: **порожній документ, стаття, книга, лист, ...**.

Для роботи із створеним проектом служать кнопка бічної панелі «**Файли та проекти**». У вікні, що відкриється, можна переглянути структуру проекту та файли, що входять до складу проекту. За його допомогою можна додати чи вилучити із проекту окремі файли. Тут також можна поміняти параметри створеного проекту.

Порядок додавання та вилучення файлів проекту


Щоби додати файл до проекту необхідно відкрити (створити) файл з документом та файл проекту. Скористатавшись контекстним меню для проекту(права клавіша миші) отримаємо:

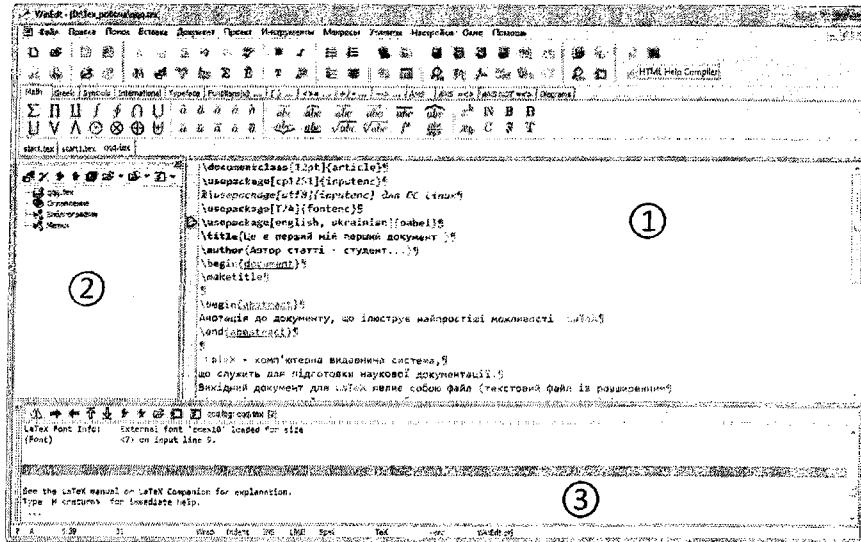


Видалення файлу з проекту не веде до його видалення і не закриває його вікна: він тільки вилучається зі списку файлів, що входять до даного проекту.

Додаток №3. Редактор WinEdt


Робоча область вікна редактора WinEdt: складається із трьох частин:


- 1 - робоче поле редактора для вводу документу;
- 2 - вікно відображення дерева проекту;
- 3 - вікно виводу інформації про хід компіляції документу. Відкривається за допомогою кнопки панелі інструментів  та відображає вміст *log* - файлу.





Вікно редактора WinEdt


Підготовлений та збережений документ за допомогою кнопок панелі інструментів можна скомпілювати та переглянути. Призначення основних кнопок (в залежності від особливостей налаштування їх графічне представлення, колір та положення на панелі інструментів) може відрізнитися від приведенного:


 (**TeXify, Ctrl+Shift+X**)- запустити процес компіляції документу. Якщо помилок на етапі компіляції виявлено не буде, то буде створено *.dvi* - файл. Інформація про хід процесу компіляції буде збережена у файлі із розширенням *.log* (у робочій папці документу із іменем, що відповідає імені вхідного документу).


 (**LaTeX, Ctrl+Shift+L**)- запустити процес компіляції документу. Якщо помилок на етапі компіляції помилок не буде, то буде створено *.dvi* - файл та запущено програму Yip для його перегляду.


 (**Компілювати виділений блок, Ctrl+Shift+C**) - запустити процес компіляції тільки виділеного у документі фрагменту. Якщо помилок на етапі компіляції виявлено не буде, то буде створено файл *_temp.pdf* (у робочій папці документу) та запущено програму для його перегляду. Документ також можна переглянути за допомогою програми *Adobe Reader*.


 (**PDF LaTeX**) - запустити процес компіляції документу. Якщо помилок на етапі компіляції виявлено не буде, то буде створено файл із розширенням *.pdf* та запущено програму для його перегляду. Результуючий документ також можна переглянути за допомогою стандартної програми для перегляду *pdf* - файлів *Adobe Reader*.


 (**Просмотр DVI, Ctrl+Shift+V**) - перегляд створеного в результаті компіляції *.dvi* -файлу.

 (**GSView, Ctrl+Shift+G**) - перегляд створеного в результаті компіляції *.pdf* -файлу (*.ps* або *.eps*) за допомогою програми *GSview*. Вона дозволяє переглядати, друкувати, міняти розміри та орієнтацію сторінок, видаляти сторінки чи переносити їх в інше місце у документі.

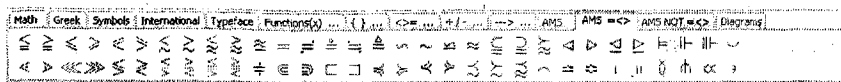
 (**Acrobat Reader**) перегляд створеного в результаті компіляції *.pdf* -файлу за допомогою програми *Acrobat Reader*.

 (**dvi2pdf**) - конвертація раніше створеного в результаті компіляції *.dvi* - файлу у *.pdf* - формат.

 (**DVIPS, Ctrl+Shift+D**) - конвертація раніше створеного в результаті компіляції *.dvi* -файлу у *.ps*-формат.

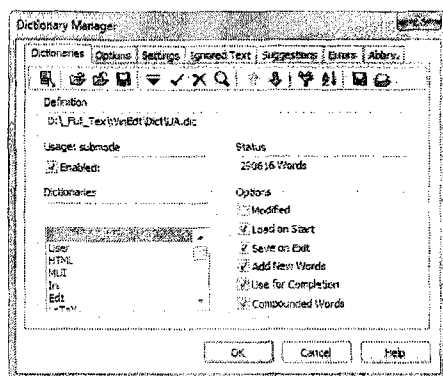
 (**ps2pdf**) - конвертація раніше створеного *.ps* - файлу у *.pdf* - формат.

Більшість операцій по введенню команд формування тексту та символів у математичних формулах може бути зведена до натискання відповідних кнопок на панелі інструментів, вибором відповідних команд за допомогою пунктів меню чи їх введенням із клавіатури. Палітра інструментів для введення основних символів та команд форматування тексту включається кнопкою Σ (**Символи TeX (панель)**), яка розгортає (згортає) приведену нижче палітру:



Словник української мови не входить до складу **WinEdt**, тому для перевірки правопису українського тексту в документі необхідно виконати наступні дії:

- розархівувати словник української мови *ua.zip* та помістити його в каталог **WinEdt/Dict/**;
- в меню **WinEdt** вибрати команду **Options/Dictionary (Налаштування/Словари)**. Відкриється менеджер словників:



- вибрати значок **Insert** (▼) та у вікні **Dictionaries** задати ім'я нового словника – **Ukrainian**. Налаштувати властивості нового словника, включивши "Enable" ("Usage submode"), "Modified", "Load On Start", "Use for Completion" ("Options");
- натиснути на піктограму "Browse" (Q), знайти та вибрати файл *ua.dic*;
- натиснути на піктограму "Update" (✓);
- натиснути на піктограму "Load" (⊞);
- для завершення роботи із менеджером словників натиснути **Ok**.

Словник буде завантажений в пам'ять редактора **WinEdt**. Аналогічним чином можна завантажити і інші словники, необхідні для роботи. Слова, що не входять до складу жодного із підключених словників, будуть виділятися іншим шрифтом: по замовчуванню це червоний підкреслений шрифт.

Додаток №4. Список команд, ключові слова в яких можна переозначити

Команда	Зміст напису (за замовчуванням)
<code>\chaptername</code>	Глава
<code>\contentsname</code>	Зміст
<code>\listfigurename</code>	Список рисунків
<code>\listtablename</code>	Список таблиць
<code>\abstractname</code>	Анотація
<code>\refname</code>	Література (для класу article)
<code>\bibname</code>	Література (для класу report і book)
<code>\indexname</code>	Предметний покажчик
<code>\figurename</code>	Рис.
<code>\tablename</code>	Таблиця
<code>\partname</code>	Частина
<code>\appendixname</code>	Додаток

Додаток №5. Таблиці спеціальних символів

Малі грецькі літери

α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>
δ	<code>\delta</code>	ϵ	<code>\epsilon</code>	ε	<code>\varepsilon</code>
ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>
ϑ	<code>\vartheta</code>	ι	<code>\iota</code>	κ	<code>\kappa</code>
λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>
ξ	<code>\xi</code>	\omicron	<code>\omicron</code>	π	<code>\pi</code>
ϖ	<code>\varpi</code>	ρ	<code>\rho</code>	ϱ	<code>\varrho</code>
σ	<code>\sigma</code>	ς	<code>\varsigma</code>	τ	<code>\tau</code>
υ	<code>\upsilon</code>	ϕ	<code>\phi</code>	ψ	<code>\psi</code>
χ	<code>\chi</code>	ψ	<code>\psi</code>	ω	<code>\omega</code>

Великі грецькі літери

Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>
Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>	Π	<code>\Pi</code>
Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>
Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>		

Символи бінарних операцій

$+$	<code>+</code>	$-$	<code>-</code>	$*$	<code>*</code>
\pm	<code>\pm</code>	\mp	<code>\mp</code>	\times	<code>\times</code>
\div	<code>\div</code>	$*$	<code>\ast</code>	\cdot	<code>\cdot</code>
\circ	<code>\circ</code>	\bullet	<code>\bullet</code>	\vee	<code>\vee</code>
\cap	<code>\cap</code>	\cup	<code>\cup</code>	\diamond	<code>\diamond</code>
\triangle	<code>\triangle</code>	∇	<code>\nabla</code>	\triangleleft	<code>\triangleleft</code>
\triangleright	<code>\triangleright</code>	\oplus	<code>\oplus</code>	\otimes	<code>\otimes</code>
\oslash	<code>\oslash</code>	\odot	<code>\odot</code>	\bigcirc	<code>\bigcirc</code>
\dagger	<code>\dagger</code>	\ddagger	<code>\ddagger</code>	\amalg	<code>\amalg</code>

Символи бінарних відношень

$>$	<code>></code>	$<$	<code><</code>	$=$	<code>=</code>
$:$	<code>:</code>	\leq	<code>\leq</code>	\geq	<code>\geq</code>
\prec	<code>\prec</code>	\succ	<code>\succ</code>	\ll	<code>\ll</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\gg	<code>\gg</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\in	<code>\in</code>
\ni	<code>\ni</code>	\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>
\equiv	<code>\equiv</code>	\models	<code>\models</code>	\sim	<code>\sim</code>
\perp	<code>\perp</code>	\simeq	<code>\simeq</code>	\mid	<code>\mid</code>
\parallel	<code>\parallel</code>	\approx	<code>\approx</code>	\cong	<code>\cong</code>
\neq	<code>\neq</code>	\propto	<code>\propto</code>	\doteq	<code>\doteq</code>

Нарядкові знаки

Нарядковий символ	LaTeX команда	Приклад
\vec{a}	<code>\vec</code>	\vec{a}
\bar{a}	<code>\bar</code>	\bar{a}
\hat{a}	<code>\hat</code>	\hat{a}
\tilde{a}	<code>\tilde</code>	\tilde{a}
\breve{a}	<code>\breve</code>	\breve{a}
\acute{a}	<code>\acute</code>	\acute{a}
\grave{a}	<code>\grave</code>	\grave{a}
\dot{a}	<code>\dot</code>	\dot{a}
\ddot{a}	<code>\ddot</code>	\ddot{a}
\dddot{a}	<code>\dddot</code>	\dddot{a}
\ddddot{a}	<code>\ddddot</code>	\ddddot{a}

Символи з межами

\bigcap	<code>\bigcap</code>	\bigotimes	<code>\bigotimes</code>	\prod	<code>\prod</code>
\bigcup	<code>\bigcup</code>	\bigsqcup	<code>\bigsqcup</code>	\coprod	<code>\coprod</code>
\bigodot	<code>\bigodot</code>	\biguplus	<code>\biguplus</code>	\sum	<code>\sum</code>
\bigoplus	<code>\bigoplus</code>	\bigvee	<code>\bigvee</code>	\int	<code>\int</code>
		\bigwedge	<code>\bigwedge</code>	\oint	<code>\oint</code>

Верхні та нижні підкреслення

Нарядковий символ	LaTeX команда	Приклад
\underline{ABC}	<code>\underrightarrow</code>	\underrightarrow{ABC}
\overline{ABC}	<code>\overrightarrow</code>	\overrightarrow{ABC}
\underline{ABC}	<code>\underleftarrow</code>	\underleftarrow{ABC}
\overline{ABC}	<code>\overleftarrow</code>	\overleftarrow{ABC}
\underline{ABC}	<code>\underleftrightharpoonup</code>	$\underleftrightharpoonup{ABC}$
\overline{ABC}	<code>\overleftrightharpoonup</code>	$\overleftrightharpoonup{ABC}$
$M = \overline{1, n}$	<code>\overline</code>	$M = \overline{1, n}$
\widehat{xyz}	<code>\widehat</code>	\widehat{xyz}
\widetilde{xyz}	<code>\widetilde</code>	\widetilde{xyz}
$\underbrace{1+3+\dots+(2n-1)}_n$	<code>\underbrace</code>	$\underbrace{1+3+\dots+(2n-1)}_n$
$\overbrace{012\dots 9AB\dots F}^{10} \overbrace{012\dots 9AB\dots F}^{10}$	<code>\overbrace</code>	$\overbrace{012\dots 9AB\dots F}^{10} \overbrace{012\dots 9AB\dots F}^{10}$

Перелік стрілок

\leftarrow	<code>\leftarrow</code> , <code>\gets</code>	\longleftarrow	\uparrow	<code>\uparrow</code>
\rightarrow	<code>\rightarrow</code> , <code>\to</code>	\longrightarrow	\downarrow	<code>\downarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	\updownarrow	<code>\updownarrow</code>
\Leftarrow	<code>\Leftarrow</code>	\Longleftarrow	\Uparrow	<code>\Uparrow</code>
\Rightarrow	<code>\Rightarrow</code>	\Longrightarrow	\Downarrow	<code>\Downarrow</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\Longleftrightarrow	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\longmapsto	\searrow	<code>\searrow</code>
\swarrow	<code>\swarrow</code>	\nearrow	\nwarrow	<code>\nwarrow</code>
\rightleftharpoons	<code>\rightleftharpoons</code>			

Роздільники

Конструкції `\left` і `\right` можна застосовувати не тільки до круглих дужок. У таблиці перераховані дужки та інші символи, які за допомогою `\left` і `\right` матимуть потрібний розмір. Їх називають роздільниками (*delimiters*).

$($	$($	\uparrow	<code>\uparrow</code>	\Uparrow	<code>\Uparrow</code>
$)$	$)$	\downarrow	<code>\downarrow</code>	\Downarrow	<code>\Downarrow</code>
$[$	$[$, <code>\lbrack</code>	\updownarrow	<code>\updownarrow</code>	\Updownarrow	<code>\Updownarrow</code>
$]$	$]$, <code>\rbrack</code>	\lfloor	<code>\lfloor</code>	\rfloor	<code>\rfloor</code>
$\{$	$\{$, <code>\lbrace</code>	\lceil	<code>\lceil</code>	\rceil	<code>\rceil</code>
$\}$	$\}$, <code>\rbrace</code>	\langle	<code>\langle</code>	\rangle	<code>\rangle</code>
\backslash	<code>\backslash</code>	$\ $	<code>\ </code>	$\ $	<code>\ </code> , <code>\Vert</code>

Спеціальні символи, що використовуються в математичних формулах

\aleph	<code>\aleph</code>	\emptyset	<code>\emptyset</code>	\forall	<code>\forall</code>
\hbar	<code>\hbar</code>	∇	<code>\nabla</code>	\exists	<code>\exists</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\neg	<code>\neg</code>
\wp	<code>\wp</code>	\bot	<code>\bot</code>	$\#$	<code>\sharp</code>
\Re	<code>\Re</code>	$\ $	<code>\Vert</code>	∞	<code>\infty</code>
\Im	<code>\Im</code>	\angle	<code>\angle</code>	\cdots	<code>\cdots</code>
∂	<code>\partial</code>	\triangle	<code>\triangle</code>	\vdots	<code>\vdots</code>
\ddots	<code>\ddots</code>				

Команди для друку імен математичних функцій

\arccos	<code>\arccos</code>	\deg	<code>\deg</code>	\sin	<code>\sin</code>
\cos	<code>\cos</code>	\ker	<code>\ker</code>	\arg	<code>\arg</code>
\csc	<code>\csc</code>	\sec	<code>\sec</code>	\coth	<code>\coth</code>
\hom	<code>\hom</code>	\tanh	<code>\tanh</code>	\exp	<code>\exp</code>
\log	<code>\log</code>	\arctan	<code>\arctan</code>	\ln	<code>\ln</code>
\tan	<code>\tan</code>	\cot	<code>\cot</code>	\sinh	<code>\sinh</code>
\arcsin	<code>\arcsin</code>	\dim	<code>\dim</code>		
\cosh	<code>\cosh</code>	\lg	<code>\lg</code>		

Додаток №6. Можливі значення параметрів для оточення *axis*

№ п/п	Параметр	Призначення
1.	xmin=значення	Початкове значення <i>x</i> для побудови графіка
2.	xmax=значення	Кінцеве значення <i>x</i> для побудови графіка
3.	ymin=значення	Початкове значення <i>y</i> для побудови графіка
4.	ymax=значення	Кінцеве значення <i>y</i> для побудови графіка
5.	ytick={y1,y2,y3,...}	Координати для відображення написів до осі <i>y</i>
6.	yticklabels={напис1,напис2,напис3,...}	Написи до осі <i>y</i>
7.	xtick={y1,y2,y3,...}	Координати для відображення написів до осі <i>x</i>
8.	xticklabels={напис1,напис2,напис3,...}	Написи до осі <i>y</i>
9.	ymax=значення	Явне задання діапазону для побудови графіка по осі ординат
10.	xmax=значення	Явне задання діапазону для побудови графіка по осі абсцис
11.	xlabel=напис	Підпис до осі <i>x</i>
12.	ylabel=напис	Підпис до осі <i>y</i>
13.	height=значення	Висота області побудови графіка
14.	width=значення	Ширина області побудови графіка

Додаток №7. Можливі значення параметрів для команди *\addplot*

№ п/п	Параметр	Призначення
1.	only marks	Відображаються тільки маркери точок
2.	mark=none	Відображаються тільки ламана лінія, що зеднує точки
3.	mark size=розмір	Розміру маркера
4.	color=значення	Колір маркера
5.	thick	Товста лінія для графіка
6.	very thick	Дуже товста лінія для графіка
7.	draw=колір	Колір лінії графіка

Додаток №8. Можливі типи маркерів для побудови графіків (для команди *\addplot*)

№ п/п	Параметр	Вигляд маркера
1.	mark=*	•
2.	mark=x	×
3.	mark=square	□
4.	mark=square*	■
5.	mark=triangle	△
6.	triangle*	▲
7.	mark=diamond	◇
8.	mark=diamond*	◆
9.	mark=pentagon	⬠
10.	mark=pentagon*	⬡




Додаток №9. Команди пакету *beamer*, що не вимагають задання міток

№ п/п	Команда та її короткий опис
1.	<code>\hyperlinkslideprev<слайди>{текст}</code> перехід на один слайд назад
2.	<code>\hyperlinkslidenext<слайди>{текст}</code> перехід на один слайд вперед
3.	<code>\hyperlinkframestart<слайди>{текст}</code> перехід на перший слайд текучого фрейму
4.	<code>\hyperlinkframeend<слайди>{текст}</code> перехід на останній слайд текучого фрейму
5.	<code>\hyperlinkframestartnext<слайди>{текст}</code> перехід на перший слайд наступного фрейму
6.	<code>\hyperlinkframeendprev<слайди>{текст}</code> перехід на останній слайд попереднього фрейму
7.	<code>\hyperlinkpresentationstart<слайди>{текст}</code> перехід на перший слайд презентації
8.	<code>\hyperlinkpresentationend<слайди>{текст}</code> перехід на останній слайд презентації
9.	<code>\hyperlinkpresentationend<слайди>{текст}</code> перехід на останній слайд презентації

Додаток №11. Команди управління переходами між слайдами

№ п/п	Команда та її короткий опис
1.	<code>\transdissolve<слайди>[параметри]</code> Зникнення попереднього слайду і його перекриття наступним. Параметр <i>duration</i> задає час переходу від одного слайду до іншого (в секундах). Наприклад <i>duration=4</i> .
2.	<code>\transblindshorizontal<слайди>[параметри]</code> Зміна слайдів горизонтальними полосами.
3.	<code>\transblindsvvertical<слайди>[параметри]</code> Зміна слайдів вертикальними полосами.
4.	<code>\transbboxin<слайди>[параметри]</code> Стиску слайду до центру та відкриття наступного.
5.	<code>\transbboxout<слайди>[параметри]</code> Розкриття наступного слайду від центру.
6.	<code>\transglitter<слайди>[параметри]</code> Ефект зміни миганням в заданому напрямі. Параметр <i>direction=315</i> задає напрям мигання.
7.	<code>\transsplithorizontalout<слайди>[параметри]</code> Зміна слайдів рухом горизонтальних полос від центру до країв.
8.	<code>\transsplitverticalout<слайди>[параметри]</code> Зміна слайдів рухом вертикальних полос від центру до країв.
9.	<code>\transsplithorizontalin<слайди>[параметри]</code> Зміна слайдів рухом горизонтальних полос від країв до центру.
10.	<code>\transsplitverticalin<слайди>[параметри]</code> Зміна слайдів рухом вертикальних полос від країв до центру.
11.	<code>\transwipe<слайди>[параметри]</code> Зміна одного слайду іншим в заданому напрямі. Параметр <i>direction</i> задає напрям заміни та може приймати значення: <i>0, 90, 180, 270</i> .
12.	<code>\transduration<слайди>{час показу}</code> Час показу текучого слайду. <i>Час показу</i> задається в секундах.

Додаток №10. Додаткові можливості малювання управляючих кнопок

№ п/п	Команда та можливе призначення кнопки	Вигляд кнопки
1.	<code>\beamergotobutton{текст}</code> перехід в іншу область презентації	
2.	<code>\beamerskipbutton{текст}</code> пропуск логічно завершеної частини тексту	
3.	<code>\beamerreturnbutton{текст}</code> Повернення до попереднього тексту	

Література

1. Кнут Д. Все про TeX. Протвино, R_DTeX, 1993 / Кнут Д. – 575 с.
2. Кнут Д. Э. Компьютерная типография / Кнут Д. – М.: Мир, 2003. – 686 с.
3. Спивак М. Восхитительный TeX / Спивак М. – М.: Мир, 1993. – 285 с.
4. Львовский С. М. Набор и верстка в пакете LaTeX. – 3-е изд. / Львовский С. М. – М.: МЦНМО, 2003. – 448 с.
5. Гуссенс М. Путеводитель по пакету LaTeX и его расширению L^AT_EX₂ ϵ / М. Гуссенс, Ф. Миттельбах, А. Самарин. – М.: Мир, 1999. – 606 с.
6. Гуссенс М. Путеводитель по пакету LaTeX и его графическим расширениям / М. Гуссенс, С. Ратц, Ф. Миттельбах. – М.: Мир, 2002. – 621 с.
7. Не очень краткое введение в LaTeX₂ ϵ , или LaTeX₂ ϵ за 94 минуты / Оетикер Т., Паргль Х., Шлегл Э., Хина И.; пер. с англ. Б. Тоботраса, 1999. – 108 с.
8. Балдин Е. М. Компьютерная типография LaTeX / Балдин Е. М. – С. Пб.: БХВ-Петербург, 2008. – 304 с.
9. Клименко С. В. LaTeX и его команды / Клименко С. В., Лисина М. В.; Институт физики высоких энергий, 1996. – 149 с.
10. Роженко А. И. Искусство верстки в LaTeX'e / Роженко А. И.; под ред. А. С. Алексеева. – Новосибирск: Изд. ИВМиМГ СО РАН, 2005. – 398 с.
11. Котельников И. А. LaTeX по-русски. – 3-е изд., перераб. и доп. / Котельников И. А., Чеботарев П. З. – Новосибирск: Сибирский хронограф, 2004. – 496 с.
12. Кренивич А. П. Издавнична система LaTeX: Методичні вказівки до лабораторних занять з дисципліни «Практикум на ЕОМ» / Кренивич А. П., Бородин В. А. – К.: ВПЦ «Київський університет», 2007. – 49 с.
13. L^Ampo_rt L. LaTeX, A Document Preparation System, User's Guide and Reference Manual, Addison-Wesley Publishing Company (1985) / L. L^Ampo_rt. – ISBN 0-201-15790-X.

ЗМІСТ

Передмова	2
Лабораторна робота №1	
Издавнична система LaTeX	3
Исторична довідка	
Теоретичні відомості	4
Хід роботи	14
Контрольні питання	15
Лабораторна робота №2	
Основи верстки документів у издавничій системі LaTeX . .	16
Теоретичні відомості	16
Хід роботи	28
Завдання для індивідуального виконання	28
Контрольні питання	33
Лабораторна робота №3	
Робота із текстом та абзацами у издавничій системі LaTeX	34
Теоретичні відомості	34
Хід роботи	45
Матеріал для створення документу	46
Завдання для індивідуального виконання	47
Контрольні питання	52
Лабораторна робота №4	
Робота зі списками у издавничій системі LaTeX	53
Теоретичні відомості	53
Хід роботи	63
Контрольні питання	64
Лабораторна робота №5	
Робота із шрифтами. Ввід простих формул	65
Теоретичні відомості	65
Хід роботи	77
Завдання для індивідуального виконання	78
Контрольні питання	82
Лабораторна робота №6	
Ввід складних формул у издавничій системі LaTeX	83
Теоретичні відомості	83
Хід роботи	93
Завдання для індивідуального виконання	93
Контрольні питання	101
Лабораторна робота №7	
Робота з таблицями у издавничій системі LaTeX	102
Теоретичні відомості	102

Хід роботи	112
Завдання для індивідуального виконання	112
Контрольні питання	114
Лабораторна робота №8	
Створення нових команд у видавничій системі LaTeX	115
Теоретичні відомості	115
Хід роботи	123
Контрольні питання	124
Лабораторна робота №9	
Графіка у видавничій системі LaTeX	125
Теоретичні відомості	125
Хід роботи	136
Завдання для індивідуального виконання	136
Контрольні питання	141
Лабораторна робота №10	
Створення комп'ютерних презентацій у видавничій системі LaTeX	142
Теоретичні відомості	142
Хід роботи	151
Завдання для індивідуального виконання	151
Контрольні питання	152
Додаток №1	
Повідомлення про помилки транслятора LaTeX	153
Додаток №2	
Редактор Kile	156
Додаток №3	
Редактор WinEdt	162
Додаток №4	
Список команд, написи в яких можна переозначити	165
Додаток №5	
Таблиці спеціальних символів	
Малі грецькі літери	166
Великі грецькі літери	166
Символи бінарних операцій	166
Символи бінарних відношень	167
Надрядкові знаки	167
Символи з межами	167
Верхні та нижні підкреслення	168
Перелік стрілок	168
Роздільники	169
Спеціальні символи	169
Команди для друку імен математичних функцій	169

Додаток №6	170
Можливі значення параметрів для оточення <i>axis</i>	
Додаток №7	
Можливі значення параметрів для команди <i>\addplot</i>	171
Додаток №8	
Можливі типи маркерів для побудови графіків	171
Додаток №9	
Команди пакету <i>beamer</i> , що не вимагають задання міток	172
Додаток №10	
Додаткові можливості малювання управляючих кнопок	172
Додаток №11	
Команди управління переходами між слайдами	173
Література	174

Навчальне видання

ТКАЧУК Валерій Михайлович

ТКАЧУК Оріся Миколаївна

ПРАКТИКУМ НА ЕОМ

ЧАСТИНА 1

ВИДАВНИЧА СИСТЕМА *LaTeX*

В авторській редакції

Головний редактор *В. М. Головчак*
Комп'ютерна верстка *В. Д. Яремко*

НБ ПНУС



785725

Підп. до друку 21.12.2012. Формат 60x84/16. Папір офсет.
Гарнітура "Times New Roman". Ум. друк. арк. 10,35.
Тираж 50 пр. Зам. № 120.

Видавець і виготовлювач
Видавництво Прикарпатського національного університету
імені Василя Стефаника
76000, м. Івано-Франківськ, вул. С. Бандери, 1
Тел. 71-56-22. E-mail: vdvcit@pu.if.ua
Свідомство суб'єкта видавничої справи ДК № 2718 від 12.12.2006