**НОРМАЛЬНІ АЛГОРИТМИ МАРКОВА: розв’язання практичних завдань**

**ТЕОРІЯ**

Підручник: В.Н. Пильщиков, В.Г. Абрамов, А.А. Вылиток, И.В. Горячая Машина Тьюринга и алгоритмы Маркова. Решение задач – сторінки 23-27

**ПРАКТИКА**

Підручник: В.Н. Пильщиков, В.Г. Абрамов, А.А. Вылиток, И.В. Горячая Машина Тьюринга и алгоритмы Маркова. Решение задач – сторінки 27-29, приклад 1-2

**Розв’язання завдань для самостійного виконання: сторінка 45.**

2.1 A={f,h,p}. В слове P заменить все пары ph на f.

Необхідно здійснити заміну всіх входжень без винятку, тому записуємо команду

Після всіх замін алгоритм зупиниться

2.2 A={f,h,p}. В слове P заменить на f только первую пару ph, если такая есть.

Оскільки команда має виконатись тільки один раз, то необхідно після її виконання завершити роботу

2.3 A={a,b,c}. Приписать слово bac слева к слову P.

Команда має виконатись один раз і здійснити зупинку, то використовуємо . Для заданого слова завжди є порожнє місце зліва і справа від початкового заданого слова. Заміна здійсниться зліва, що нам і потрібно

2.4 A={a,b,c}. Заменить слово P на пустое слово, т.е. удалить из P все символы.

Відкинемо всі букви а, тоді всі b, тоді всі c

2.5 A={a,b,c}. Заменить любое входное слово на слово a.

Відкинемо всі букви, поставимо один раз «а» і завершимо роботу.

2.6 Построить НАМ, не меняющий входное слово (при любом алфавите A).

2.7 A={ | }. Считая слово P записью числа в единичной системе счисления, получить остаток от деления этого числа на 2, т.е. получить слово из одной палочки, если число нечётно, или пустое слово, если число чётно.

Одинична (унітарна) система числення задає число через необхідну кількість |. Відкидатимемо по дві палочки, якщо одна палочка залишилась, то вона не мала пари, і кількість палочок непарна, якщо всі в парах, то в залишку нічого не буде, порожнє слово

2.8 A={ | }. Считая слово P записью положительного числа в единичной системе счисления, уменьшить это число на 1.

Необхідно одну палочку відкинути і зупинитись

2.9 A={ | }. Считая слово P записью числа в единичной системе счисления, увеличить это число на 2.

До порожньої позиції зліва додаємо дві палочки і зупиняємося

2.10 A={0,1,2}. Считая слово P записью числа в троичной системе счисления, получить остаток от деления этого числа на 2, т.е. получить слово 1, если число нечётно, или слово 0, если число чётно. (Замечание: в чётном троичном числе должно быть чётное количество цифр 1.)

Відкидаємо всі нулі і двійки, які не відіграють жодної ролі і парності

Залишиться послідовність одиниць, перевіряємо її на парність згідно задачі 2.7

Якщо кількість одиниць непарна, то залишиться одна одиниця, що є відповіддю, її потрібно зробити кінцевою , щоб не перейти на останню команду. Якщо третя команда знищить всі пари одиниць і їх не залишиться, алгоритм пропустить четверту команду, виконає п’яту команду, поставить нуль і зупиниться.

2.11 A={a,b,c}. Определить, входит ли символ a в слово P. Ответ (выходное слово): слово a, если входит, или пустое слово, если не входит.

Відкинемо всі букви «б» і «с». Залишиться ряд букв «а». Зменшимо їх до однієї букви, замінивши дві букви на одну. Остання команда буде виконуватись до тих пір поки не залишиться одна єдина буква а.

Якщо нема жодної букви «а», то відповіддю буде порожнє слово.

2.12 A={a,b}. Если в слово P входит больше символов a, чем символов b, то в качестве ответа выдать слово из одного символа a, если в P равное количество a и b, то в качестве ответа выдать пустое слово, а иначе выдать ответ b.

Спочатку посортуємо : спочатку букви а , потім б. . Відкидатимемо пари «». Якщо кількість букв а більша, то залишиться тільки букви а, необхідно зменшити кількість «а» до однієї, як в попередній задачі, якщо більше «», то зменшити «».

Перша команда буде виконуватись до тих пір, поки вона може знайти сполучення тобто перша команда посортує: поміняє місцями всі букви , які стоять перед . Тільки після того як не буде можливості здійснити першу команду виконується друга команда.

**Зауваження для всіх задач**: Алгоритм завжди виконує команди з початку. Якщо виконались всеможливі варіанти першої команди, то ОДИН раз здійснюється друга команда і знову алгоритм біжить на початок: можливо перша команда стала доступною. І т.д.

2.13 A={0,1,2,3}. Преобразовать слово P так, чтобы сначала шли все чётные цифры (0 и 2), а затем – все нечётные.

Здійснимо сортування: якщо трійка стоїть перед парною цифрою то поміняти їх місцями, якщо 1 стоїть перед парними – теж саме

**Д/З**

2.14 A={a,b,c}. Преобразовать слово P так, чтобы сначала шли все символы a, затем – все символы b и в конце – все символы c.

2.15 A={a,b,c}. Определить, из скольких различных символов составлено слово P; ответ получить в единичной системе счисления (например: acaac → | | ).

А = {a, b, с}. Составить НАМ, перемещающий все буквы «a» в конец слова

**НОРМАЛЬНІ АЛГОРИТМИ МАРКОВА-2**

**ПРАКТИКА.**

**Частина 2**

Підручник: В.Н. Пильщиков, В.Г. Абрамов, А.А. Вылиток, И.В. Горячая Машина Тьюринга и алгоритмы Маркова. Решение задач – сторінки 29-36, приклад 3-6

**Блок 1. Розв’язання завдань для самостійного виконання: сторінка 46.**

2.16 A={a,b,c}. В непустом слове P удвоить первый символ, т.е. приписать этот символ слева к P.

Спочатку розібрати приклад 3 на сторінці 29.

Використаємо спец символ \*. Введення спецсимволу буде останньою командою у програмі

\* стоятиме біля першого символу, саме його треба подвоїти і зупинитись.

**Зауваження**: Якщо існує ймовірність порожнього початкого слова, то алгоритм необхідно доповнити командою виходу , що не відбулось зациклення

Д/З 2.17 A={a,b,c}. За первым символом непустого слова P вставить символ c.

2.18 A={a,b,c}. Из слова P удалить второй символ, если такой есть.

Використаємо спец символ \*.

\* буде біля першого символу. Перший символ залишаємо без змін і для того, щоб відзначити, що ми пройшли перший символ, змінюємо спец символ на #. Ми наче перескочили через перший символ і стоїмо # перед другим символом

Залишається тільки «проковтнути» другий символ.

Врахуємо що другого символу може не бути, як може і не бути першого символу, тоді :

2.19 A={a,b,c}. Если в слове P не менее двух символов, то переставить два первых символа.

Можна розв’язати методом перебору ймовірних варіантів (Д/З продумайте інший спосіб)

Якщо перші і друга букви однакові, тоді їх можна не переставляти місцями

Залишається тільки врахувати випадок коли другої букви нема, і жодної букви нема. Оскільки ми перебрали можливі варіанти, то це буде попадати під команду випадок

2.20 A={0,1,2}. Считая непустое слово P записью троичного числа, удалить из этой записи все незначащие нули.

Всі нулі з початку будемо знищувати, дійшовши до значущої цифри (1 або 2)

\* проковтне початкові нулі та самознищиться.

2.21 A={a,b,c}. Приписать слово abc справа к слову P.

Приклад 5 ст. 34 підправити.

2.22 A={a,b,c}. Удалить из непустого слова P его последний символ.

\* біжить в кінець слова

І в кінці стоятиме після останньої букви, яку треба знищити.

Якщо необхідно було б врахувати ймовірність порожнього початкового слова, то додаємо

Д/З 2.23 A={0,1}. Считая непустое слово P записью числа в двоичной системе, получить двоичное число, равное учетверённому числу P (например: 101 → 10100).

2.24 A={0,1}. Считая непустое слово P записью числа в двоичной системе, получить двоичное число, равное неполному частному от деления числа P на 2 (например: 1011 → 101).

2.25 A={a,b}. В слове P все символы a заменить на b, а все (прежние) символы b – на a

В цій задачі теж слід використовувати спец символ, щоб пройтись від початку до кінця і всі букви поміняти. Одразу проходячи ми наче перескочили букву, помінявши її на протилежну та йдемо далі. Та додаємо команду стоп-кран

**Блок 2. Спробуємо розв’язати в НАМ задачі,що зустрічали в МТ**

**Унітарна система числення:**

1). Побудувати НАМ для функції x + y

Дано два числа в унітарній системі. Між ними розділювач «+». Знайти суму

111+11

Фактично достатньо забрати +

Питання: чи потрібно команду робити завершальною?

Інший спосіб розв’язання з розділювачем #

<https://phm.cuspu.edu.ua/theory_of_algorithms/markov_1.html>

2). Побудувати НАМ для функції x – y

Поки-що розглянемо випадок, коли *х* більше *у.* Розділювач «-»

*111111-111*

Достатньо забирати по одній одиниці з кожного числа і залишати «-»

Залишилось тільки забрати «-»

Можна покрочно перевірити виконання <https://phm.cuspu.edu.ua/theory_of_algorithms/markov_2.html>

Символ «^»- це порожній символ в даному проекті

Перевірте виконання заданого НАМ, поставивши значення *у* більше *х* в текстовому рядку «Вхідне (вихідне) слово:». Яка команда в НАМ враховує цей випадок?

3). Побудувати НАМ для функції x / 2 (х-парне)

Дано парне число. Необхідно пробігтися від початку до кінця (використати спец символ \*) і замість двох одиниць за собою залишати одну . Та не забуваємо забрати \* і зупинити алгоритм, щоб другий раз не виконати останню команду.

<https://phm.cuspu.edu.ua/theory_of_algorithms/markov_3.html>

Як діятиме алгоритм у випадку непарного числа?

4). Побудувати НАМ для знаходження остачі від ділення на 3 для заданого в унітарній системі числення числа. Відповідь задати в десяткові системі.

Тобто для числа 8: 11111111 – остача ділення на 3 буде дорівнювати 2.

Відкинемо по три 1, які не впливають на відповідь

Може залишитись одна, або дві одиниці або жодна. Обов’язково розглядаємо у випадку спадання: дів одиниці замінимо числом 2, одні -1, жодну – нулем

Якщо залишити команди без змін, то остання команда зациклить алгоритм (попередня у окремих випадках теж), тому кожну команду робимо кінцевою.

**Десяткова система числення**

5). Дано число. Знайти наступне

Необхідно пройти в кінець заданого числа і здійснювати зміни з кінця

Коли ми дійшли до самого кінця символ \* можна замінити на #, який буде здійснювати збільшення числа в розряді у випадках 0, 1, …8 і зупинятись;

у випадку «9» буде рухатись вліво до наступного розряду

Врахуємо випадок 999, тоді # поставить замість 9 нулі і опиниться на початку заданого числа, отже 1., щоб отримати 1000.

Питання: яка з двох команд і є лишньою?6). Записати остачу від ділення на 2.

Йдемо в кінець числа і по останній цифрі (\* стоятиме біля останньої цифри) пишемо остачу. І включаємо символ #

Після цього # повинен знищити всі цифри, що знаходяться перед ним. Сам # ліквідовується і закриває роботу алгоритму

**ІНДИВІДУАЛЬНІ ЗАВДАННЯ**

**Виконати три завдання + індивідуальне завдання згідно порядкового номеру в журналі**

|  |  |
| --- | --- |
|  | Задано число в десятковій системі числення. Записати МТ, що обчислює остачу від ділення на десять. |
|  | Дано число в десятковій системі числення. Знайти попереднє число |
|  | Задано число в системі числення з основою 5. Збільшити число на 2 |
|  | Дано число в десятковій системі числення, більше нуля. Відкинути в ньому всі парні цифри (0,2,4,6;8) |
|  | Задано число в системі числення з основою 7. Записати МТ для знаходження наступного числа |
|  | Дано число в системі числення 5. Знайти наступне число |
|  | Дано послідовність одиниць. Утворити послідовність одиниць, розширивши попередню, поставивши після кожної одиниці нуль (було 111 – стало 101010) |
|  | Дано число в десятковій системі числення. Чи число ділиться на 5. Так- відповідь 1, ні – відповідь 0 |
|  | Задано число в десятковій системі числення. Записати МТ, що обчислює остачу від ділення на десять |
|  | Дано число в унітарній системі числення, сказати чи воно ділиться на 3 |
|  | Алфавіт {а,б,с}. Визначити чи в початковому слові кількість букв «а» у ділиться на 3. Відповідь: 1 – так, 0 – ні |
|  | Дано число в унітарній системі числення, збільшити його втричі |
|  | Задано число в системі числення з основою 7. Записати МТ для знаходження наступного числа . |
|  | Алфавіт {а,б,с}. Визначити чи кількість букв «с» у слові парна. Відповідь: 1 – так, 0 – ні |
|  | Дано число в унітарній системі числення, зменшити його втричі |
|  | Задана послідовність одиниць. Побудувати нову послідовність, яка після двох одиниць ставить 0. |
|  | Алфавіт {а,б,с}. З початкової послідовності утворити нову в якій збережені, тільки кожен другий елемент (будо «абсабс» – стало «бас») |
|  | Дано число в двійковій системі числення сказати чи воно парне |
|  | Задана послідовність одиниць. Взначити чи їх кількість кратна 5. Якщо кратна відповідь 1, якщо ні -0. |
|  | Задано число в системі числення з основою 10. Збільшити число на 3 |
|  | Задана послідовність одиниць. Побудувати нову послідовність, яка втричі більша за попередню. |
|  | Алфавіт {а,б,с}.Чи є в початковому слові підряд хоча б дві букви а. Відповідь: 1 – так, 0 – ні |
|  | Дана послідовність одиниць. Залишити тільки ту кількість одиниць, що ділиться на 3. Ті одиниці, що утворюють остачу відкинути. (було 1111111 – стало 111111) |
|  | Алфавіт {а,б,с}. Визначити чи кількість букв у слові парна. Відповідь: 1 – так, 0 – ні |

**НОРМАЛЬНІ АЛГОРИТМИ МАРКОВА-3**

**ПРАКТИКА.**

**Частина 3**

Підручник: В.Н. Пильщиков, В.Г. Абрамов, А.А. Вылиток, И.В. Горячая Машина Тьюринга и алгоритмы Маркова. Решение задач – сторінки 29-36, приклад 3-6

**Блок 1. Розв’язання завдань для самостійного виконання: сторінка 46.**

2.27 A={a,b}. Приписать слева к слову P столько палочек, сколько всего символов входит в P (например: babb → ||||babb).

Пройдемо від початку слова до кінця, кожну букву будемо «перескакувати» і залишати саму букву і паличку біля неї

Буде стільки палочок скільки є символів. Тепер залишилось перемістити всі палочки вліво. Це можна здійснити простим сортуванням

Залишається організувати стоп-кран, щоб остання команда не виконалась вдруге і не запустила перші команди .

2.28 A={a,b}. Пусть слово P имеет чётную длину (0, 2, 4, …). Удалить левую половину этого слова. (Рекомендация: использовать решение предыдущей задачи.)

Повністю беремо попередній розв’язок

Зліва знаходяться палички, справа слова з буквами.

Нам потрібно забрати половину букв, тому ми беремо дві палички та одну букви і «викидаємо їх»

Палочки закінчаться вдвічі швидше, букв пропаде тільки половина

2.29 A={a,b}. Пусть длина слова P кратна 3. Удалить правую треть этого слова

Повністю по аналогії: ставимо палочки, сортуємо палочки тепер в праву сторону, забираємо три палочки і одну букву.

2.30 A={a,b}. Приписать справа к слову P столько палочек, со скольких подряд идущих символов a начинается это слово (например: aababa → aababa| | ).

Ставимо на початок слова \*. Якщо слово починається з букв «с» або «b», то взагалі нічого не потрібно робити і ці команди ми не прописуємо. Якщо слово починається з «а», то в нас є випадок \*а, ми перескакуємо букву, залишивши ще палочку. Якщо є ще «а», то команда повториться.

Сортуємо палочки вправо

Залишається оформити зупинку

2.31 A={a,b,c}. Удалить из слова P второе вхождение символа a, если такое есть.

Ідея: Скориставшись \* йдемо від початку слова по буквам, ми пропускаємо букви «b» i «с», якщо зустрінемо «а», то міняємо знак \* на інший, наприклад #, який дальше йтиме по слові до кінця.

Якщо # зустріне «а», то дана буква буде другою буквою «а» по рахунку і її треба знищити і зупинити роботу, а якщо зустріне інші букви, то пропускати і йти дальше.

Необхідно врахувати, що в слові може не бути другої букви «а» і треба # знищити, і може бути випадок, що слово взагалі не містить букви «а».

2.32 A={a,b,c}. Удалить из слова P третье вхождение символа a, если такое есть.

Аналогічно до попередньої задачі вводять додаткові символи, в задачі 2.31 їх було два \* і #. Тут їх необхідно три. Третій спецсимвол «з’їсть» третю букву.

2.33 A={a,b,c}. Оставить в слове P только первое вхождение символа a, если такое есть.

Задача подібна до задачі 2.31. \* буде залишати першу букву «а», в той час як # буде всі інші букви а знищувати.

2.34 A={a,b,c}. В непустом слове P оставить только последний символ.

Ідея: йдемо вкінець символом \*, перескакуємо останній елемент, включаємо символ #, який проковтне всі букви які є перед ним.

Д/З 2.35 A={a,b,c}. Из всех вхождений символа a в слово P оставить только последнее вхождение, если такое есть.

2.36 A={a,b,c}. Если слово P начинается с символа a, то заменить P на пустое слово, а иначе P не менять.

Якщо є випадок \*а, то запускаємо # для знищення слова, інші випадки (інші букви) будуть стоп-краном.

Дані команди , можна не писати, бо це окремі випадки команди , яка в нас є.

2.37 A={a,b}. Если слово P содержит одновременно символы a и b, то заменить P на пустое слово.

Якщо в початковому слові міститься словосполучення «ab» або «ba» то запускаємо «вірус» у вигляді спецсимволу &, що знищувати все навколо себе

Залишилось знищити всі осередки вірусу

Чи можна останню команду замінити на ?

Продумайте ще способи розв’язання даної задачі.

2.38 A={a,b,c}. Если буквы в непустом слове P не упорядочены по алфавиту, то заменить P на пустое слово, а иначе P не менять.

Якщо є три символи, то у випадку не відсортованості можливі наступні випадки: ba, ca, cb. Нехай ці випадки запускають вірус, що знищить всі довкола. Точок «спалаху вірусу» може бути декілька

Якщо букви посортовані, то жодна команда не виконається.

**ІНДИВІДУАЛЬНІ ЗАВДАННЯ**

|  |  |
| --- | --- |
|  | A = {0,1,2,3}. Вважаючи непорожнє слово P записом четверичного числа, отримати залишок від ділення цього числа на 4 |
|  | А={a,b,c}. Визначити чи входить в нього хоча б два символи «а». Відповідь-1, ні- 0. |
|  | A = {0,1,2}. Вважаючи непорожнє слово P записом трійкового числа, збільшити це число на 1 |
|  | А={a,b,c}. Подвоїти кожен символ |
|  | А={a,b,c}. Якщо в початкове слово не входить символ «а», то замінити всі букви «b» на «с». В іншому випадку залишати без змін |
|  | A = {0,1,2,3}. Вважаючи непорожнє слово P записом четверичного числа, перевірити, чи є воно парним. Відповідь: слово 0, якщо число є парним, і слово 1, якщо число є непарним. |
|  | А={a,b,c}. Замінити на «с» кожен третій символ в початковому слові |
|  | А={a,b,c}. Визначити чи є в початковому слові словосполучення abc. Якщо є то відповідь 1 , якщо ні -0 |
|  | А={a,b,c}. Знищити два останні символи |
|  | А={a,b,c}. Знищити третій символ з кінця |
|  | A = {0,1,2}. Вважаючи непорожнє слово P записом позитивного трійкового числа, зменшити це число на 1. |
|  | А={a,b}. Визначити чи входить в нього ще раз його перший символ. Відповідь-1, ні- 0. |
|  | {a,b,0,1} . Визначити чи задане початкове слово є записом числа в двійковій системі числення. Відповідь: 1- так, 0- ні |
|  | A = {a,b}. Визначити, чи входить перший символ непорожнього слова P ще раз в це слово. Відповідь: слово a , якщо входить, або порожнє слово, якщо не входить. |
|  | А={a,b,c}. Знищити перший та останній символи |
|  | А={a,b,c}. Якщо в початкове слово входить символ «а», то замінити всі букви «b» на «с». В іншому випадку залишати без змін |
|  | А={a,b,c}. Визначити чи є в початковому слові більше двох букв «с». Відповідь: 1- так, 0- ні |
|  | А={a,b,c}. Третю букву з кінця замінити на «с» |
|  | А={a,b,c}. Замінити на «а» кожен другий символ в заданому слові |
|  | А={a,b,c}. Якщо в початковому слові нема підряд двох букв «с», то знищити все слово |
|  | А={a,b,c}. Знищити третю букву «с» |
|  | А={a,b,c}. Кожен другий символ знищити |
|  | Записане число в трійковій системі числення. Визначити чи воно є парним (кількість цифр 1 парна для парних чисел) |
|  | А={a,b,c}. Всі символи на непарних позиціях подвоїти |