Група ІПС-42, Гончарук Ангеліна Віталіївна

Тема: Алгоритм Евкліда. Варіант 27

Завдання.

Знайти d - найбільший спільний дільник чисел a,b,

розв'язати рівняння au+bv=d,

розв'язати рівняння as+bt=c,

Для прикладів 1 і 2 навести всі кроки алгоритму, для прикладу 3 можна навести тільки 5 перших і 5 останніх кроків.

4. Розробити програму і провести чисельний експеримент для дослідження часової складності алгоритму. Навести результати експерименту для прикладів. Навести псевдокод вибраного Вами алгоритму.

5. Навести теоретичний матеріал по обчисленню часової складності для трьохстрічкової машини Тьюрінга.

1. a = 1365

b = 406

c = 14

Відповідь:

d = 7

u = -11

v = 37

s = -22

t = 74

2. a = 32505055

b = 5215298

c = 9361

Відповідь:

d = 407

u = -3873

v = 24139

s = -89079

t = 555197

3. a = 3045510812896021

b = 708377096971726

c = 119662

Відповідь:

d = 47

u = -3635013947301

v = 15627939312068

s = -9254745509828346

t = 39788733488525130

**Розв’язування**

**1. Прямий хід**

1365 = 3 ∙ 406 + 147

406 = 2 ∙ 147 + 112

147 = 1 ∙ 112 + 35

112 = 3 ∙ 35 + 7**(d = 7)**

35 = 5 ∙ 7 + 0

**Зворотний хід**

147 = 1 ∙ a - 3 ∙ b

112 = -2 ∙ a + 7 ∙ b

35 = 3 ∙ a - 10 ∙ b

7 = -11 ∙ a + 37 ∙ b

**(u = -**11**, v =** 37**)**

**2. Прямий хід**

32505055 = 6 ∙ 5215298 + 1213267

5215298 = 4 ∙ 1213267 + 362230

1213267 = 3 ∙ 362230 + 126577

362230 = 2 ∙ 126577 + 109076

126577 = 1 ∙ 109076 + 17501

109076 = 6 ∙ 17501 + 4070

17501 = 4 ∙ 4070 + 1221

4070 = 3 ∙ 1221 + 407**(d=407)**

1221 = 3 ∙ 407 + 0

**Зворотний хід**

1213267 = 1 ∙ a + -6 ∙ b

362230 = -4 ∙ a + 25 ∙ b

126577 = 13 ∙ a + -81 ∙ b

109076 = -30 ∙ a + 187 ∙ b

17501 = 43 ∙ a + -268 ∙ b

4070 = -288 ∙ a + 1795 ∙ b

1221 = 1195 ∙ a + -7448 ∙ b

407 = -3873 ∙ a + 24139 ∙ b

**(u =** -3873**, v =** 24139**)**

**3.Прямий хід**

3045510812896021 = 4 ∙ 708377096971726 + 212002425009117

708377096971726 = 3 ∙ 212002425009117 + 72369821944375

212002425009117 = 2 ∙ 72369821944375 + 67262781120367

72369821944375 = 1 ∙ 67262781120367 + 5107040824008

67262781120367 = 13 ∙ 5107040824008 + 871250408263

∙∙∙

105233 = 13 ∙ 7990 + 1363

7990 = 5 ∙ 1363 + 1175

1363 = 1 ∙ 1175 + 188

1175 = 6 ∙ 188 + 47**(d=47)**

188 = 4 ∙ 47 + 0

**Зворотний хід**

708377096971726 = 0 ∙ a + 1 ∙ b

212002425009117 = 1 ∙ a + -4 ∙ b

72369821944375 = -3 ∙ a + 13 ∙ b

67262781120367 = 7 ∙ a + -30 ∙ b

5107040824008 = -10 ∙ a + 43 ∙ b

871250408263 = 137 ∙ a + -589 ∙ b

∙∙∙

7990 = -6392831792 ∙ a + 27484567797 ∙ b

1363 = 87567417677 ∙ a + -376476764329 ∙ b

1175 = -444229920177 ∙ a + 1909868389442 ∙ b

188 = 531797337854 ∙ a + -2286345153771 ∙ b

47 = -3635013947301 ∙ a + 15627939312068 ∙ b

**(u =** -3635013947301**, v = 15627939312068)**

4. Програма була реалізована на мові програмування Python 3.8.

Тестування було проведено на наборі пар чисел, що не є взаємно простими та еквівалентиними. Позначимо функцію знаходження НСД розширеним алгоритмом Евкліда:

НСД(a, b)

Результати тестування кожної пари чисел записуються у файл для подальшої репрезентації на графіку.

**Псевдокод:**

НСД(а, b):

доки b != 0:

a, b = b, a mod b

повернути a

знайти\_тестову\_пару(початок\_інтервалу, кінець\_інтервалу):

а = -1

b = -1

доки Істина:

а := випадкове\_ціле(початок\_інтервалу, кінець\_інтервалу)

b := випадкове\_ціле(початок\_інтервалу, кінець\_інтервалу)

g = НСД(а, b)

якщо a != b і a != g і b != g і g != 1:

повернути (a, b)

запустити\_тестування():

m := 100

для кожного \_ з проміжку (10, 1000):

a, b := знайти\_тестову\_пару(ціле(m), ціле(m\*2))

записати\_результат(виміряти\_час(a, b))

m := m \* 1.2

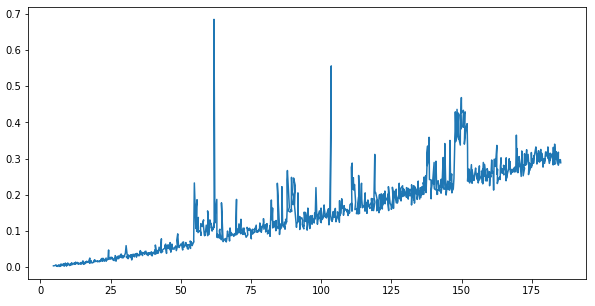
виміряти\_час(a, b):

початок := час\_зараз()

r := НСД(a, b)

повернути час\_зараз() – початок

Результати тестування відображені на шкалою синім кольором.



**t, c**

**log(min(a, b))**

На вертикальній осі відображений час виконання тесту пари. 5.

Алгоритм

1. A<-N, b<-M

2. t<-a-b|; якщо t=0, зупинка

3. a<-t, b<-min{a, b}; перехід на 2

Маши́на Тю́рінга — математичне поняття, введене для формального уточнення інтуїтивного поняття алгоритму. У кожної машини Тюрінга є стрічка, потенційно нескінченна в обидві сторони. Є скінченна множина символів стрічки S0, …, Sn, що називається алфавітом машини. У кожен момент часу кожна комірка може бути зайнята не більш ніж одним символом. Машина має деяку скінченну множину внутрішніх станів q0, q1, …, qn. У кожен даний момент часу машина знаходиться лише в одному із цих станів.

Нарешті, є голівка, яка у кожен даний момент часу знаходиться на одній із комірок стрічки. Машина діє не безупинно, а лише у дискретні моменти часу. Якщо у якийсь момент t голівка сприймає комірку (тобто знаходиться на комірці), що містить символ Si, і машина знаходиться у внутрішньому стані qj, то дія машини визначена однією із чотирьох дій:

* голівка затирає символ Si, і записує у тій же комірці новий символ Sk,
* голівка пересувається в сусідню ліву комірку,
* голівка пересувається в сусідню праву комірку,
* машина зупиняється.

У випадках (1)-(3) машина переходить у новий внутрішній стан qr, і готова знову до дії у наступний момент t + 1. Припустимо, що символ S0 представляє порожню комірку, і отже, голівка завжди сприймає деякий символ.

Перші три з можливих дій машини можуть бути описані відповідно такими упорядкованими четвірками, які називаються командами:

1. Si qj -> Sk qr S
2. Si qj -> Sk qr L
3. Si qj -> Sk qr R

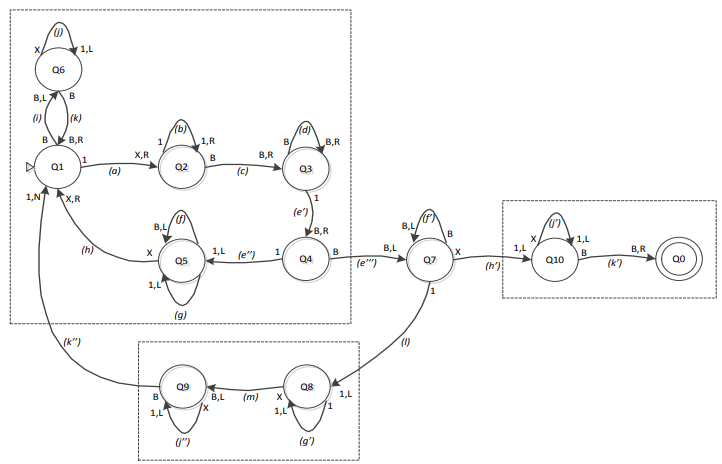
Тут перші два символи — це відповідно внутрішні стани машини і сприйнятий символ, наступні три визначають дію машини (запис голівкою символу Sk, або переміщення голівки на одну комірку вліво, або переміщення голівки на одну комірку вправо) та новий стан машини. Якщо стрічка вкладена в машину Тюрінга, голівка машини встановлена на одну із комірок стрічки і машина приведена в один із своїх внутрішніх станів, то машина починає оперувати на стрічці: її голівка пише і стирає символи і пересувається з однієї комірки в іншу. Якщо при цьому машина колись зупиняється, то стрічка, яка розташована в ній в цей момент, називається результатом застосування машини до даної стрічки.

Незважаючи на свій простий устрій, машина Тюрінга може виконувати складні перетворення слів. Спочатку, коли стрічка містить вхідне слово, автомат знаходиться проти деякої з комірок і у якомусь стані. У залежності від вибору початкової комірки, утворяться різні результати роботи машини Тюрінга. У процесі своєї роботи машина Тюрінга буде ніби перестрибувати з однієї комірки програми на іншу, відповідно до інформації на стрічці і вказівками в командах програми, поки не дійде до команди, яка переведе автомат у кінцевий стан. Детермінована машина Тюрінга має функцію переходу, яка по комбінації поточного стану і символу на стрічці визначає три речі: символ, що буде записаний на стрічці, напрямок зміщення головки по стрічці і новий стан скінченного автомату. Наприклад, X на стрічці в стані 3 однозначно визначає перехід в стан 4, запис на стрічку символу Y і ​​переміщення головки на одну позицію вліво. ДМТ із трьома стрічками називається четвірка M = (K, X, δ, 𝑠0), де K – скінченна множина, елементи якої називаються станами ДМТ, Х – скінчення множина, попарно різних елементів, що називається алфавітом ДМТ (причому 𝐾 ∩ 𝑋 = ∅), а алфавіт Х включає символ # (пустий символ), ⊳ (початковий символ) і пусте слово ⅇ) , 𝑠0 - початковий стан, функція переходів δ, яка називається програмою, визначається наступним чином:

δ ∶ K × X3 → (𝐾 ∪ {ℎ, "yes", "no"}) × (𝑋 × {𝑙, 𝑟,𝑡})3

На першій стрічці, яка називається вхідною, записується вхідне слово, з цієї стрічки дозволяється лише читати і не дозволяється нічого записувати. Друга стрічка, робоча, або ж внутрішня, така, що на неї можна писати і з неї можна читати. На третю стрічку, яка називається вихідною, дозволяється тільки писати. Розв'язання проблеми такої машиною виконується так. На вхідній стрічці записується вхідне слово в алфавіті Х={0,1, ⊳, #} починаючи з найлівішої клітинки (тобто з першої клітинки стрічки), а в решту клітинок стрічки записаний символ # - символ пустої клітинки. Машина працює з цим словом відповідно до своєї програми (функції переходів), записує необхідні символи на вихідну і робочі стрічки. Запис на вихідну стрічку теж виконується з самої лівої клітинки, а в решту клітинок стрічки записаний символ #. Якщо в деякий момент машина зупиняється з певною відповіддю, то вона записує цю відповідь – “yes” або “no” - на вихідну стрічку.

Побудова машини



Часовою складністю розв’язання проблеми називається число кроків тристрічкової ДТМ, які вона виконала до зупинки. Пам’яттю, або простором обчислень називається число клітинок робочої стрічки, які використала ДТМ під час обчислень. Мова L ⊆ (𝑋 ∪ {#})∗ належить класу часової складності *TIME(f(n))*, якщо існує трьохстрічкова ДМТ, яка розв’язує мову *L* в часі *f(n).* Множину мов *TIME(f(n))* називають класом часової складності.