

Олімпіада з computer science

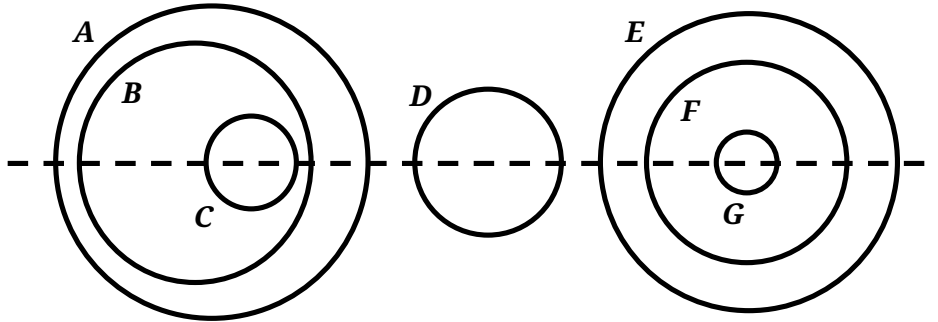
2 травня 2017 р.

Максимальна оцінка за кожну з трьох задач — 14 балів

Час виконання роботи — 4 години

1. Кола

На площині накреслено декілька кіл. Центри усіх кіл лежать на одній прямій — осі абсцис, — і ніякі два кола не перетинаються, тобто не мають жодної спільної точки. *Ступенем відрізаності одне від одного* двох кіл із цього набору назвемо мінімальну кількість кіл, які потрібно прибрати, щоб між даними двома колами можна було провести ламану, яка не перетинає жодного з кіл, що лишилися. Наприклад, на рисунку нижче ступінь відрізаності кола C від кола F складає 3: необхідно прибрати кола A , B та E .



Завдання 1. Припустімо, конкретне розташування кіл нам не відоме, але відомим є набір точок, у яких кола перетинають вісь абсцис, та для кожної точки ми знаємо, є вона лівою чи правою межею деякого кола. Доведіть, що за цією інформацією можна однозначно відновити початкове розташування кіл.

Завдання 2. Нехай тепер точки перетину кіл із віссю абсцис (разом з інформацією для кожної точки, є вона лівою чи правою межею кола) задано в порядку зростання координат. Покажіть, як за лінійний від загальної кількості кіл час¹ знайти пару кіл максимальної відрізаності одне від одного (якщо таких пар є кілька, то довільну).

¹ Повинно існувати таке стає число C , що для довільної кількості N кіл та для довільного розташування цих кіл кількість операцій, які виконує алгоритм, не перевищує CN .

2. Сортувальний автомат

У перших N клітинках нескінченної в один бік стрічки записано додатні цілі числа; у решті клітинок — нулі. У першій клітинці стоїть автомат, який вміє тримати в пам'яті одне-єдине число. Спочатку це число дорівнює нулю. *Інструкцією* назвимо вказівку автомату виконати одну з трьох дій:

- перейти на одну клітинку праворуч;
- перейти на одну клітинку ліворуч;
- обміняти вміст клітинки, в якій зараз стоїть автомат, і вміст внутрішньої пам'яті, тобто стерти попереднє число в клітинці, записати туди число, яке зберігається в автоматі, після чого записати в пам'ять число, яке раніше стояло у клітинці.

Набором інструкцій назвемо послідовність таких інструкцій (можливо, різних) у чітко заданому порядку і чітко заданої довжини.

Програмою назвемо функцію, що приймає на вхід число N , номер клітинки, у якій стоїть автомат, та число, записане в цій клітинці, а також число, записане на даний момент у пам'яті автомата, а повертає набір інструкцій, який необхідно виконати автомату. Якщо набір інструкцій має ненульову довжину, автомат його виконує, після чого знову викликається функція-програма, що визначає подальші дії автомата; якщо на якомусь етапі повернений набір інструкцій має нульову довжину, автомат припиняє свою роботу.

Тривалістю виконання програми назвемо сумарну кількість інструкцій, виконаних автоматом до моменту завершення роботи.

Завдання 1. Опишіть таку програму, що після завершення роботи автомата числа в перших N клітинках стрічки буде розташовано в порядку неспадання (і набір цих чисел збігатиметься, за винятком порядку, з початковим). Автомат має рано чи пізно завершувати свою роботу для довільного початкового набору чисел.

Завдання 2. Опишіть програму, що задовольняє умову завдання 1 та є найефективнішою (для найгіршого випадку) серед таких програм². Відповідь обґрунтуйте. Найефективнішими можуть бути водночас кілька програм; вам достатньо навести лише один приклад.

² Під максимальною ефективністю програми P -Best для найгіршого випадку ми розуміємо те, що для довільної іншої програми P , яка задовольняє умову, існує таке стає число C , що для довільного N існує такий набір вхідних чисел S -Worst, що для довільного набору S , який складається з N чисел, кількість операцій, які виконує автомат P -Best для набору S не більше ніж у C разів перевищує кількість операцій, які автомат P виконує для набору S -Worst.

3. Найбільший спільний дільник

Алгоритм Евкліда пошуку найбільшого спільного дільника (НСД) двох невід’ємних цілих чисел A та B (хоча б одне з яких не дорівнює нулю) можна описати так: якщо одне з чисел нуль, слід повернути інше число з пари, інакше — продовжити виконання алгоритму для нової пари чисел, одне з яких дорівнює меншому з чисел A та B , а друге — остачі від ділення на нього більшого числа. Наприклад, для чисел 3700 і 999 пошук найбільшого спільного дільника (він дорівнює 37) виглядатиме таким чином:

$$(3700, 999) \rightarrow (999, 703) \rightarrow (703, 296) \rightarrow (296, 111) \rightarrow (111, 74) \rightarrow (74, 37) \rightarrow (37, 0) \rightarrow 37$$

Завдання 1. Доведіть, що кількість операцій, які виконує алгоритм Евкліда, є в найгіршому випадку логарифмічною³ від суми вхідних чисел.

Завдання 2. Часто комп’ютери ефективніше виконують ділення на 2, ніж операцію ділення у загальному випадку. Вкажіть ще один логарифмічний алгоритм пошуку НСД, який, однак, не виконує операцій ділення, крім, якщо потрібно, ділення на число 2. Доведіть, що алгоритм дійсно є логарифмічним від суми вхідних чисел.

³ Логарифмом числа x за основою u називається число z , для якого $u^z = x$. Позначається такий логарифм як $\log_u(x)$. Наприклад, $\log_3(81) = 4$, $\log_2(2048) = 11$, а $\log_{10}(1234) \approx 3,09$. Тут ми просимо довести, що існують такі сталі додатні числа C і c , що кількість операцій, які виконує алгоритм Евкліда для пари початкових чисел A та B (якщо $A + B > 1$) ніколи не є більшою за число $C \times \log_2(A + B)$, але існують пари чисел (A, B) із як завгодно великою сумою, для яких кількість виконуваних операцій не є меншою за $c \times \log_2(A + B)$.