

XIX Всеукраїнська олімпіада з інформатики

Другий тур

Пошук рядку

У послідовності, що складається з маленьких символів латинського алфавіту, потрібно знайти підпослідовність найбільшої довжини, яка складається з різних символів, що йдуть підряд у послідовності.

Завдання

Напишіть програму SUBSTR, що за заданою послідовністю знаходить першу підпослідовність, яка складається з різних символів.

Вхідні дані

Вхідний файл SUBSTR.DAT містить послідовність, яка, для зручності, розбита на декілька рядків. Кожен рядок містить не більше 100 символів. Загальна довжина послідовності – не більше 10 000 000 символів.

Вихідні дані

Єдиний рядок вихідного файлу SUBSTR.SOL має містити першу з підпослідовностей найбільшої довжини, що не містить однакових символів.

Приклад вхідних та вихідних даних

SUBSTR.DAT	SUBSTR.SOL
abcabcdabc	abcd
bacdbca	

Щасливий білет

Білет на планеті Олімпія містить шестизначний номер. Мешканці планети дещо незвичайно визначають чи є білет *щасливим*. Вони загадують деяке число k , а потім намагаються скласти з номера це число за допомогою таких правил.

- Спочатку номер розбивають на цифри, та певні сусідні цифри об'єднують у числа.
- Поміж отриманих чисел розташовують скобки та знаки операцій: плюс, мінус, умножити, розділити за математичними правилами. Також дозволяється використовувати унарний мінус.
- Неможна міняти місцями цифри.
- Якщо вдається скласти такий вираз, який після обчислення дорівнюватиме шуканому числу k , то білет вважається *щасливим*.

Операцію «розділити» можна використовувати тільки у випадках, коли в наслідок ділення буде отримано ціле число.

Наприклад, розглянемо білет номер: 182836. Візьмемо $k=840$. Розіб'ємо число на чотири: 1, 8, 2, 836. Число k можна отримати, наприклад, таким чином: $1*(8/2+836)=840$.

Завдання

Напишіть програму LUCKY, що за номером білету та числом k визначить чи є цей номер *щасливим*, та знайде один з варіантів розбиття номеру на послідовність чисел, між якими можна розташувати математичні операції та скобки для отримання шуканого числа k .

Вхідні дані

Єдиний рядок вхідного файлу LUCKY.DAT містить два числа: ціле k ($1 \leq k \leq 1000$) та номер білету. Номер білету складається з шести цифр, та може починатися з 0.

Вихідні дані

Єдиний рядок вихідного файлу LUCKY.SOL повинен містити будь-який з можливих наборів чисел на які може бути розбитий номер білету для отримання числа k . Якщо число не може бути отримане, то єдиний рядок вихідного файлу повинен містити число 0 (нуль).

Приклади вхідних та вихідних даних

LUCKY.DAT	LUCKY.SOL
840 182836	1 8 2 836
120 000001	0

Відстань

На площині своїми координатами задано N точок. Розглянемо набір прямих, проведених через усі різні пари точок. Необхідно визначити найбільшу можливу відстань від будь-якої заданої точки, до будь-якої прямої побудованої за двома іншими точками.

Завдання

Напишіть програму DIST, що за переліком точок площини обчислює максимальну відстань від точки до прямої.

Вхідні дані

Перший рядок вхідного файлу DIST.DAT містить єдине ціле число – кількість точок N ($3 \leq N \leq 700$) заданих на площині. Далі йдуть N рядків, кожен з яких задає точку площини у форматі "x y" ($-5000 \leq x, y \leq 5000$), x та y – цілі числа. Жодні дві точки не мають однакових координат.

Вихідні дані

Єдиний рядок вихідного файлу DIST.SOL повинен містити найбільшу відстань від однієї з заданих точок, до прямої побудованої на двох інших точках, з точністю до 10^{-6} . Відповідь повинна бути записана у форматі з точкою (<ціла частина>.<дрібна частина>)

Приклад вхідних та вихідних даних

DIST.DAT	DIST.SOL
5	4.242641
1 4	
2 0	
2 4	
3 5	
4 4	

