**Новорічна гра**

Ім’я вхідного файлу: game.in

Ім’я вихідного файлу: game.out

На новорічному святі Дід Мороз вирішив провести цікаву гру для двох найвеселіших дітлахів.

У мішку Діда Мороза знаходяться не подарунки, а ***N*** карток, на кожній із яких написано деяке ціле число ***Ai***. У гру грають двоє дітей. На початку гри обидва гравці навмання витягують по одній картці з мішка Діда Мороза. Учасник, який вийняв картку з більшим числом, отримує від Діда Мороза цукерки, до того ж кількість отриманих цукерок дорівнює різниці чисел, написаних на картках, що вийняли діти.

Наприклад, Сергій та Роман грають у цю гру. Сергій вийняв картку з числом 7, а Роман з числом 3. Після цього Сергій бере собі 4 (7 - 3 = 4) цукерки.

Дід Мороз стомився і не може визначити, скільки цукерок йому слід купити на свято. Допоможіть йому. Його цікавить максимальна кількість цукерок, яку може отримати дитина в результаті гри.

***Формат вхідних даних:*** перший рядок вхідного файлу містить одне ціле число ***N(2 ≤ N ≤ 1000)***, яке представляє кількість карток.

Другий рядок вхідного файлу містить рівно ***N*** цілих чисел ***Ai(1 ≤ Ai ≤ 32767)***. Числа у рядку розділені одиночними пробілами. ***Ai*** – число, написане на ***і***-й картці.

***Формат вихідних даних:*** єдиний рядок вихідного файлу має містити одне число – максимальна кількість цукерок, яку може отримати дитина за гру.

***Приклад вхідних і вихідних даних:***

|  |  |
| --- | --- |
| **game.in** | **game.out** |
| 27 3 | 4 |
| 54 2 7 9 5 | 7 |
| 43 3 3 3 | 0 |

**ACM World Finals**

Ім’я вхідного файлу: acm.in

Ім’я вихідного файлу: acm.out

Дехто з вас, мабуть, знає, що кожного року проводиться чемпіонат світу з програмування серед студентів. У фінал цього змагання проходять близько 80 команд з усього світу.

Кожна команда складається з трьох чоловік і має назву. Напишіть програму, яка по короткій назві команди і прізвищах її учасників, формує повну назву команди.

Повна назва команди складається з короткої назви команди і списку прізвищ її учасників. Прізвища учасників у списку мають бути впорядковані за абеткою і відділені одне від іншого комами. Назва команди від прізвищ учасників має бути відділена двокрапкою. Після кожного розділового знаку має бути рівно один пробіл.

***Формат вхідних даних:*** вхідний файл містить рівно 4 рядки. Перший рядок містить назву команди. Кожен із наступних трьох рядків містить прізвище одного із членів команди. Довжини рядків не перевищують 50 символів.

***Формат вихідних даних:*** єдиний рядок вихідного файлу має містити рівно один рядок з повною назвою команди.

***Приклад вхідних і вихідних даних:***

|  |  |
| --- | --- |
| **acm.in** | **acm.out** |
| Dream TeamKnuthDijkstraCormen | Dream Team: Cormen, Dijkstra, Knuth |

**Сума дільників**

Ім’я вхідного файлу: sum.in

Ім’я вихідного файлу: sum.out

Нехай ***х*** – натуральне число. Назвемо число ***y*** його дільником, якщо ***1 ≤* у ≤ х** і залишок від ділення **x** на **y** дорівнює нулю.

Задано число ***х***. Знайдіть суму його дільників.

***Формат вхідних даних:*** перший рядок вхідного файлу містить одне ціле число **х** ***(1* ≤ х ≤ *1018)***. Усі прості дільники числа ***х*** не перевищують 1000.

***Формат вихідних даних:*** єдиний рядок вихідного файлу має містити одне число – суму дільників заданого числа ***х***.

***Приклад вхідних і вихідних даних:***

|  |  |
| --- | --- |
| **sum.in** | **sum.out** |
| 12 | 28 |
| 239 | 240 |

**Вписаний трикутник**

Ім’я вхідного файлу: triangle.in

Ім’я вихідного файлу: triangle.out

Дано опуклий ***N***-кутник. Знайдіть площу максимального трикутника, який лежить всередині (або на межі) цього ***N***-кутника.

***Формат вхідних даних:*** у першому рядку записано число ***N*** – кількість вершин многокутника ***(3 ≤ N ≤ 2000)***. Далі йдуть ***N*** рядків, у кожному з яких записані два цілих числа, що не перевищують по модулю 107 - координати вершини многокутника. Координати йдуть в порядку обходу многокутника або за часовою стрілкою, або проти.

***Формат вихідних даних:*** виведіть площу знайденого трикутника з точністю до трьох знаків.

***Приклад вхідних і вихідних даних:***

|  |  |
| --- | --- |
| **triangle.in** | **triangle.out** |
| 40 010 012 120 10 | 70.000 |

**Подорож**

Ім’я вхідного файлу: travel.in

Ім’я вихідного файлу: travel.out

Маленький Степан дуже любить подорожувати. У країні Берляндії, де він живе, є ***N*** міст, розташованих на одній прямій. Степан пронумерував їх номерами від 1 до ***N*** в порядку збільшення краси. Степан перебуває у місті 1 і хоче потрапити в місто ***N***. Щоб не псувати враження про поїздку, він може відвідувати міста тільки в порядку збільшення номерів (а, отже, і краси).

Для переміщення між містами Степан вирішив скористатися послугами єдиної авіакомпанії країни – Berland Airlines. Вартість перельоту з міста ***i*** в місто ***j*** дорівнює ***ci*\*|*xi-xj*|+*tj***, де ***xi*** *–* координата міста ***i*, *xj*** – координата міста ***j***, ***ci*** – вартість одиниці літакового палива в місті ***i***, а ***tj*** – вартість в’їзду у місто ***j***. Щоб було про що розповісти друзям, Степан хоче потратити як можна більше (так-так, саме більше) грошей на цю подорож. Допоможіть йому в цьому.

Напишіть програму, яка за заданими ***N***, координатам міст, вартостям палива в містах, а також вартостям в’їзду в міста знайде максимальну кількість грошей, які Степан може витратити, щоб добратися з міста 1 до міста ***N***. Зауважте, йому не обов’язково бувати в усіх містах.

***Формат  вхідних даних:*** перший рядок вхідного файлу містить ціле число ***T*** - кількість тестів ***(1 ≤ T ≤ 3)***. Перший рядок кожного тесту містить ціле число ***N*** – кількість міст в Берляндії ***(1*** ***≤ N ≤*** ***100000)***. Далі йдуть ***N*** рядків. Рядок з номером ***i*** з них містить 3 цілих числа –   ***xi*** ***(-106*** ***≤*** ***xi*** ***≤ 106)***, ***ci* *(1 ≤*** ***ci*** ***≤ 106)*** и ***ti* *(1 ≤ ti* *≤ 106)***. Усі координати – різні.

***Формат  вихідних даних:*** для кожного тесту в окремому рядку виведіть шукану найбільшу кількість грошей, які Степан може витратити щоб дістатися з міста 1 до міста ***N***. Гарантується, що відповідь не перевищує 1012.

***Приклад вхідних і вихідних даних:***

|  |  |
| --- | --- |
| **travel.in** | **travel.out** |
| 145 10 20 1 1015 3 1417 2 3 | 123 |

# *Пояснення:* нижче показані два можливих оптимальних маршрути:

 1 → 4, 1 → 3 → 4.

***Оцінювання:***

1. У наборі тестів загальною вартістю 30 балів ***N*** не перевищує 2000.
2. У наборе тестів загальною вартістю 55 балів координати міст зростають при зростанні номера.
3. Тести, у яких виконуються обидві умови 1 і 2, мають вартість 15 балів.
4. Тести, у яких виконується хоча б одна із перших двох умов, мають вартість 70 балів.