

Задача А. Тура

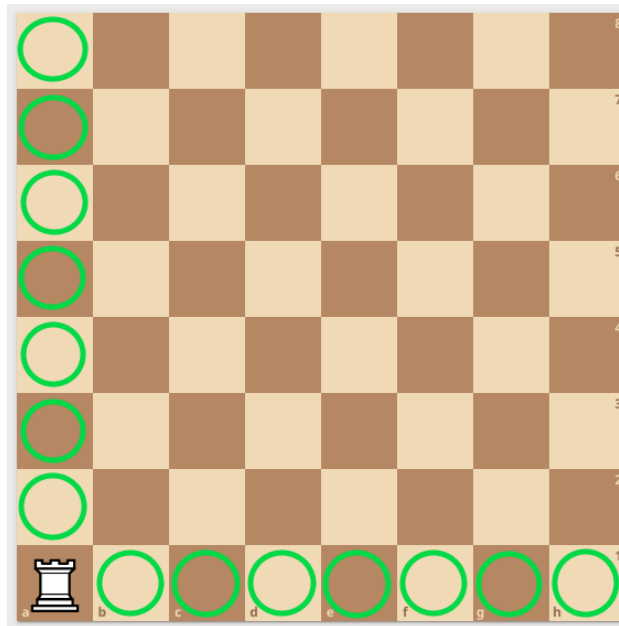
Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано шахівниця розміром $n \times m$. Тобто з n рядками та m стовпчиками.

У цій шахівниці є лише одна фігура — тура. Вона знаходиться у нижньому лівому куті. Більше ніяких фігур немає.

Нагадаємо, що тура за один хід може переміститися на будь-яку кількість клітин по горизонталі або вертикалі, але не по діагоналі.

Знайдіть кількість клітин, на які тура може переміститися за один хід.



На малюнку зображена традиційна шахівниця розміру 8×8 . У ній тура може переміститися на всі клітини, які помічені зеленим. Таких всього 14, тому відповідь 14.

Формат вхідних даних

Перший рядок містить два цілі числа n та m ($1 \leq n, m \leq 20$) — розміри шахівниці.

Формат вихідних даних

Виведіть кількість клітин, на які тура може переміститися за один хід.

Приклади

standard input	standard output
8 8	14
3 2	3

Зауваження

Пояснення, чому до першого прикладу відповідь 14, можна побачити на малюнку вище.

У другому прикладі відповідь 3, бо тура може переміститися лише на одну позицію вгору та на дві позиції вправо.

Задача В. Кава

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: `1 second`
Ліміт використання пам'яті: `256 megabytes`

Андрій вирішив випити кави в одній Київській кав'ярні. Проте він згадав, що у Києві "червона зона".

Нагадаємо, що у "червоній зоні" у Києві лише повністю вакциновані люди (тобто ті, які отримали два щеплення) можуть відвідувати кав'ярні. Якщо ж людина неповнолітня (строго менше ніж 18 років), то вона може відвідати кав'ярню лише з повністю вакцинованим дорослим, при цьому дитина не зобов'язана бути вакцинованою.

Андрію n років та він отримав m щеплень. А його повнолітній батько, Борис, отримав уже k щеплень.

Визначте, чи зможе Андрій потрапити у кав'ярню. Можливо, разом з батьком.

Формат вхідних даних

Перший рядок містить одне ціле число n ($12 \leq n \leq 30$) — вік Андрія.

Другий рядок містить два цілі числа m та k ($0 \leq m, k \leq 2$) — кількість щеплень, які отримали Андрій та Борис відповідно.

Формат вихідних даних

Виведіть «Yes», якщо Андрій зможе потрапити у кав'ярню, або «No» — інакше.

Ви можете виводити букви у будь-якому регістрі.

Приклади

standard input	standard output
15 1 2	Yes
29 1 2	No

Задача С. Коробка

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

У Козака Вуса є коробка, яка може вмістити до k кілограмів включно. Якщо у коробці будуть речі, вага яких перевищує k кілограмів, то вона порветься. У нього також є п'ять м'ячів вагою a_1, a_2, a_3, a_4, a_5 кілограмів відповідно. Також відомо, що вага кожного наступного м'яча більша за попередню.

Визначте максимальну кількість м'ячів, які можна положити у коробку так, що вона не порвалася.

Формат вхідних даних

Перший рядок містить одне ціле число k ($1 \leq k \leq 100$).

Другий рядок містить п'ять цілих чисел a_1, a_2, a_3, a_4, a_5 ($1 \leq a_i \leq 25$). Гарантується, що кожне наступне число більше за попереднє.

Формат вихідних даних

Виведіть максимальну кількість м'ячів, які можна вмістити у коробку.

Приклади

standard input	standard output
10 1 4 5 7 25	3
4 1 2 3 4 5	2
9 1 2 3 4 5	3

Зауваження

У першому прикладі перші три речі сумарно важать 10 кілограмів, саме стільки можна вмістити у коробку.

У другому прикладі перші дві речі важать три кілограми. А три речі важать уже шість кілограмів, проте шість більше, ніж чотири. Тому третю річ взяти неможливо.

У третьому прикладі перші три речі важать шість кілограмів, а чотири речі важать уже десять кілограмів, тобто більше, ніж дев'ять. Тому відповідь три.

Задача D. Яблука

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

У Козака Вуса є нескінченна кількість зелених, жовтих та червоних яблук.

Він буде їх виставляти в один ряд за наступним алгоритмом:

Спочатку g зелених яблук, потім y жовтих, потім r червоних, потім y жовтих, потім g зелених, а потім все знову нескінченну кількість разів. Наприклад, якщо $g = 1$, $y = 3$, $r = 4$, то перші яблука виглядатимуть так:

`GYYYRRRRYYYGGYYRRRRYYYGGYYRRR...`

Тобто, всі яблука можна розбити на блоки. На цьому малюнку один блок це `GYYYRRRRYYYG`. Зверніть увагу, що блок починається і закінчується на один й той же символ. Саме тому на першому малюнку йдуть дві `G` підряд попри те, що $g = 1$.

Знайдіть колір яблука, що знаходиться на n -ій позиції.

Формат вхідних даних

Перший рядок містить чотири цілі числа n, g, y, r ($1 \leq n, g, y, r \leq 10^{12}$).

Зверніть увагу, що числа не вміщуються у 32-бітні цілочисельні типи даних. Тому потрібно використовувати 64-бітні цілочисельні типи даних, такі як `long long` в C++, `long` в Java чи `int64` в Pascal.

Формат вихідних даних

Якщо колір буде зелений, то виведіть «G».

Якщо колір буде жовтий, то виведіть «Y».

Якщо колір буде червоний, то виведіть «R».

Приклади

standard input	standard output
1 1 3 4	G
2 1 3 4	Y
12 1 3 4	G
13 1 3 4	G
20 1 3 4	R
1000000000000 1 3 4	Y

Задача Е. Максимум

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Є масив a довжини $n + 1$. Проте ви його не знаєте.

Дано масив b довжини n , де $b_i \geq \max(a_i, a_{i+1})$. Тобто b_i більший або рівний максимуму з a_i та a_{i+1} .

Знайдіть максимально можливу суму чисел масиву a .

Формат вхідних даних

Перший рядок містить одне ціле число n ($1 \leq n \leq 100$).

Другий рядок містить n цілих чисел b_1, b_2, \dots, b_n ($1 \leq b_i \leq 1000$).

Формат вихідних даних

Виведіть одне ціле число — відповідь на задачу.

Приклади

standard input	standard output
2 2 3	7
3 3 2 3	10
3 1 7 1	4

Зауваження

У першому прикладі масив може виглядати так $[2, 2, 3]$.

У другому прикладі масив може виглядати так $[3, 2, 2, 3]$.

У третьому прикладі масив може виглядати так $[1, 1, 1, 1]$. Зверніть увагу, що $b_2 \geq \max(a_2, a_3)$, тобто $7 \geq \max(1, 1)$. Зверніть увагу, що там знак \geq , а не $=$.

Задача F. Прямі

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано n точок на декартовій системі координат. Тобто кожна точка має координати (x, y) .

Знайдіть кількість трійок точок, які знаходяться на одній горизонтальній або вертикальній прямій. Тобто, потрібно порахувати кількість таких трійок (a, b, c) , що $1 \leq a < b < c \leq n$ та p_a, p_b, p_c — на одній прямій, де p_i — i -та точка.

Для 50% тестів точок рівно три.

Формат вхідних даних

Перший рядок містить одне ціле число n ($3 \leq n \leq 100$).

Кожен з наступних n рядків містить по два цілі числа x_i та y_i ($1 \leq x_i, y_i \leq 1000$) — координати i -ої точки.

Гарантується, що всі точки різні.

Формат вихідних даних

Виведіть кількість трійок точок, що знаходяться на одній прямій.

Система оцінки

Ваш розв'язок отримує принаймні 50% балів, якщо воно буде правильно працювати для $n = 3$.

Приклади

standard input	standard output
6 1 1 1 2 1 3 2 2 2 3 3 3	2
3 5 6 5 3 5 10	1

Зауваження

У першому прикладі є дві трійки точок, що знаходяться на одній прямій — це трійки $[(1, 1), (1, 2), (1, 3)]$ та $[(1, 3), (2, 3), (3, 3)]$. Зверніть увагу, що трійка $[(1, 1), (2, 2), (3, 3)]$ не рахується через те, що вона формує пряму по діагоналі, а нам потрібні лише ті, які формують або горизонтальні прямі, або вертикальні.

У другому прикладі є одна трійка $[(5, 6), (5, 3), (5, 10)]$.

Задача G. Операції

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано n цілих чисел a_1, a_2, \dots, a_n . Спочатку вони всі рівні нулю.

Дано m операцій, кожен з яких описується двома числами k_i та c_i , які означають, що ви можете k_i разів вибрати будь-який елемент з масиву a та замінити його значення на c_i . Зверніть увагу, що елементи, які ви вибираєте, не обов'язково мають бути різними. Також ви не зобов'язані робити i -ту операцію рівно k_i разів, ви можете виконати її будь-яку кількість разів, але не більше k_i . Також ви можете не виконувати операцію взагалі.

Всі m операцій ви маєте виконувати послідовно. Тобто, спочатку всі заміни першої операції, потім другої, і так далі.

Знайдіть максимальну можливу суму масиву, що може вийти в кінці.

Формат вхідних даних

Перший рядок містить два цілі числа n та m ($1 \leq n, m \leq 10^5$).

Кожен з наступних m рядків містить по два цілі числа k_i та c_i ($1 \leq k_i, c_i \leq 10^5$).

Формат вихідних даних

Виведіть одне ціле число.

Приклади

standard input	standard output
3 2 2 1 2 3	7
10 1 6 3	18

Задача Н. Тура, але уже складніша

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 2 seconds
Ліміт використання пам'яті: 256 megabytes

Пам'ятаєте першу задачу? Ця задача дуже схожа.

Дано шахівниця розміром $n \times m$. Тобто з n рядками та m стовпчиками.

У цій шахівниці є фігура — тура. Вона знаходиться у нижньому лівому куті, яка має координати $(1, 1)$. Протилежний кут має координати (n, m) . Також дано k інших фігур. i -та фігура має координати (x_i, y_i) , де $1 \leq x_i \leq n$, $1 \leq y_i \leq m$.

Порахуйте кількість клітин, на які може переміститися тура не більше, ніж **за два ходи**. Зверніть увагу, що позицію, на які зараз тура, рахувати непотрібно. Тура не може бити інші фігури, а також не може перескакувати через них. Інші фігури не рухаються.

Формат вхідних даних

Перший рядок містить три цілі числа n, m, k ($1 \leq n, m \leq 2 \cdot 10^5$, $0 \leq k \leq 2 \cdot 10^5$).

Кожен з наступних k рядків містить два цілі числа x_i та y_i ($1 \leq x_i \leq n$, $1 \leq y_i \leq m$). Гарантується, що всі фігури, включно з турою, знаходяться на різних позиціях.

Формат вихідних даних

Виведіть одне ціле число.

Система оцінки

У 60% тестів виконуються обмеження $n, m \leq 1000$.

У 80% тестів виконуються обмеження $n, m \leq 10000$.

Приклади

standard input	standard output
3 3 2 2 3 3 2	5
3 3 0	8
4 4 2 1 2 2 1	0
4 5 2 3 1 2 4	14

Зауваження

У першому прикладі можна потрапити у клітини $[(1, 2), (1, 3), (2, 1), (2, 2), (3, 1)]$.

У другому прикладі можна потрапити в усі клітини.

У третьому прикладі не можна потрапити у жодну клітину.

Четвертий приклад пояснений нижче. Тут T — тура. X — фігура. 1 — позиція, яку можна досягти. 0 — позиція, яку неможливо досягти.

```
0 1 1 0 1
X 1 1 0 1
1 1 1 X 1
T 1 1 1 1
```