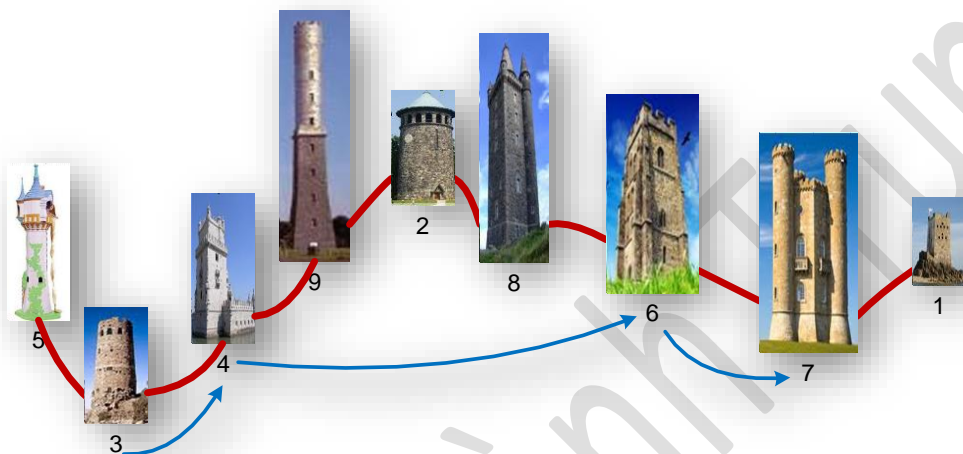


TUYẾN PHÒNG VỆ

*Tên file: DEFENSE.****

Sau cuộc chiến tranh gần nhất, đất nước bị tàn phá nặng nề. Quốc vương xứ Ardenia quyết định củng cố tuyến phòng thủ thủ đô. Tuyến phòng thủ bao gồm dãy các pháo đài nằm trên đường nối thủ đô với khu rừng ở phía bắc, bao gồm n pháo đài đánh số từ 1 đến n kể từ pháo đài gần thủ đô nhất, pháo đài thứ i có độ cao h_i ($1 \leq h_i \leq 10^9$, $1 \leq n \leq 10^3$, $i = 1 \div n$). Quốc vương yêu cầu các quân sư đánh giá khả năng phòng thủ. Các quân sư trình bày khá nhiều và lâu, nhưng nhà vua chỉ nắm được là chất lượng tuyến phòng thủ phụ thuộc vào dãy con các pháo đài có độ cao tăng dần tính từ thủ đô (do đặc trưng cách vây đánh của kẻ địch).



Ví dụ, với $n = 9$ và độ cao các pháo đài là (5, 3, 4, 9, 2, 8, 6, 7, 1) thì dãy các pháo đài với độ cao (3, 4, 6, 7) sẽ quyết định chất lượng của toàn tuyến phòng thủ.

Yêu cầu: Cho n và các h_i . Hãy xác định số lượng pháo đài quyết định chất lượng của toàn tuyến phòng thủ.

Dữ liệu: Vào từ file văn bản DEFENSE.INP:

- Dòng đầu tiên chứa số nguyên t – số lượng tests ($1 \leq t \leq 25$),
- Mỗi test cho trên 2 dòng:
 - Dòng thứ nhất chứa số nguyên n ,
 - Dòng thứ 2 chứa n số nguyên h_1, h_2, \dots, h_n .

Kết quả: Đưa ra file văn bản DEFENSE.OUT, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên – số lượng pháo đài quyết định chất lượng của toàn tuyến phòng thủ.

Ví dụ:

DEFENSE.INP										
2										
9										
5	3	4	9	2	8	6	7	1		
7										
1	2	3	10	4	5	6				

DEFENSE.OUT	
4	
6	



d14 CERC 10 D

Dãy con có tổng bằng S**Tên file: SUMS.CPP**

Cho dãy gồm n số nguyên dương a_1, a_2, \dots, a_n và một giá trị S .

Hãy chọn ra trong dãy một dãy con (không nhất thiết liên tiếp) có tổng bằng S .

Input: SUMS.INP

- Dòng 1 chứa số nguyên dương n và S ($1 \leq n \leq 100, 1 \leq S \leq 10000$)
- Dòng 2 chứa n số nguyên dương a_1, a_2, \dots, a_n ($1 \leq a_i \leq 100$)

Output: SUMS.OUT

- Nếu có thể chọn ra được dãy con có tổng bằng S thì:
 - + Dòng đầu ghi thông báo: "YES"
 - + Dòng thứ hai đưa ra dãy các vị trí của dãy con có tổng bằng S
- Ngược lại thì thông báo "NO"

Ví dụ:

SUMS.INP	SUMS.OUT
7 9	YES
3 4 2 1 4 2 3	1 3 4 7

Xâu con chung dài nhất**Tên file: LCS.CPP**

Xâu con của xâu X thu được bằng cách xóa đi một vài ký tự của X và giữ nguyên vị trí của các ký tự còn lại. Ví dụ: 'abc' là xâu con của xâu 'adcberc', và không phải là xâu con của xâu 'adcber'.

Cho hai xâu ký tự là X và Y .

Tìm xâu con chung có độ dài lớn nhất của hai xâu X và Y .

Input: LCS.INP

- Dòng 1 xâu X
- Dòng 2 xâu Y
- Cả 2 xâu X và Y có số lượng ký tự không quá 100

Output: LCS.OUT

- Dòng 1: xâu con chung dài nhất của hai xâu X, Y
- Dòng 2: các vị trí trong xâu X của xâu con chung dài nhất
- Dòng 3: các vị trí trong xâu Y của xâu con chung dài nhất

LCS.INP	LCS.OUT
AGTXAGT	GXT
GAXTA	2 4 7
	1 3 4

Túi ba gang

Tên file: CAYKHE.***

Trong truyện cổ tích "Cây Khế" ta đã biết rằng chim thần chở người anh với một cái túi ba gang đến hòn đảo đầy vàng bạc châu báu. Người em băn khoăn không biết chọn đồ vật nào cho vào túi vì chỉ có một cái túi ba gang..

Giả sử rằng trên hòn đảo kia có N đồ vật khác nhau, đồ vật thứ i có giá trị là a_i và có thể tích là b_i . Cũng giả sử rằng cái túi mà người em mang đi chỉ có thể tích là M . Bạn hãy giúp người em chọn ra trong N đồ vật trên một số đồ vật sao cho tổng thể tích của các đồ vật được chọn không vượt quá M và tổng giá trị các đồ vật được chọn là lớn nhất.

Input: Cho trong file văn bản CAYKHE.INP

- Dòng đầu tiên ghi hai số N, M ($N, M \leq 100$)
- N dòng tiếp theo, dòng thứ i ghi hai số a_i và b_i lần lượt là giá trị và thể tích của đồ vật thứ i ($a_i, b_i \leq 100$)

Output: Ghi ra file văn bản CAYKHE.OUT

- Dòng đầu tiên ghi tổng giá trị lớn nhất có thể cho vào trong túi
- Dòng thứ hai ghi số hiệu các đồ vật được cho vào trong túi. Đầu tiên ghi K là số lượng đồ vật được chọn, tiếp theo là K số thể hiện số hiệu các đồ vật được chọn.

Ví dụ:

CAYKHE . INP	CAYKHE . OUT
5 10	63
20 3	3 1 2 4
19 1	
30 7	
24 3	
15 6	

Xếp va ly**Tên file: VALY.CPP**

Một va ly có thể chứa W đơn vị trọng lượng. Có n đồ vật mỗi vật có trọng lượng $A[i]$ và có giá trị $C[i]$ hỏi lên chọn mỗi loại đồ vật bao nhiêu để xếp vào valy sao cho tổng giá trị của valy là lớn nhất.

Dữ liệu vào: valy.inp

- Dòng đầu tiên là số N và W ($1 \leq N, W \leq 100$)
- Dòng thứ i trong số N dòng tiếp theo mỗi dòng ghi hai số $A[i]$ và $C[i]$ ($1 \leq A[i], C[i] \leq 100$)

Kết quả ra : file valy.out

- Dòng đầu tiên là tổng giá trị lớn nhất tìm được của valy
- Các dòng tiếp theo mỗi dòng ghi hai số i (là số hiệu vật được chọn) x (là số lượng chọn vật i)

Ví dụ

valy.inp	valy.out
4 10	108
5 4	2 2
1 9	3 1
8 90	
2 16	

Biến đổi xâu ký tự**Tên file: changest.*****

Với một xâu ký tự S cho trước, ta có thể thực hiện các phép biến đổi sau:

- D: Xoá một ký tự của xâu S . Ký hiệu $D i$ trong đó i là vị trí cần xóa
- I: Chèn trước vị trí t của xâu S một ký tự c nào đó. Ký hiệu $I t c$. Qui định thêm về vị trí chèn: nếu xâu S có độ dài k , vị trí chèn là $1, 2, 3, \dots, k+1$, chèn ở vị trí $k+1$ có nghĩa là viết thêm vào cuối xâu S
- R: Thay ký tự thứ t của S bởi ký tự c nào đó. Ký hiệu $R t c$

Giả sử X và Y là hai xâu ký tự. Độ dài xâu X là n , độ dài xâu Y là m ($0 \leq m, n \leq 100$)

Hãy tìm một dãy gồm ít nhất các phép biến đổi biến xâu X thành xâu Y (số phép biến đổi ít nhất này gọi là khoảng cách giữa hai xâu)

Dữ liệu vào cho trong file CHANGEST.INP gồm hai dòng

- Dòng thứ nhất là xâu X
- Dòng thứ hai là xâu Y

Kết quả ghi ra file CHANGEST.OUT:

- Dòng thứ nhất ghi số K , đó là khoảng cách giữa hai xâu
- K dòng tiếp theo mỗi dòng ghi ký hiệu một phép biến đổi theo trình tự thực hiện để biến X thành Y

Ví dụ:

CHANGEST . INP	CHANGEST . OUT
ertrtyui	6
tyuhj	D 1
	D 1
	D 1
	D 1
	I 4 h
	R 5 j

Dãy chung dài nhất của hai dãy số

Tên file: COMSEQ.***

Cho hai số nguyên dương M, N ($0 < M, N \leq 1000$) và hai dãy số nguyên: a_1, a_2, \dots, a_m (A) và b_1, b_2, \dots, b_n (B). Tìm một dãy dài nhất C nhận được từ A bằng cách xoá đi một số số hạng và cũng nhận được từ B bằng cách xoá đi một số số hạng (nói chung không cùng chỉ số như đối với dãy A). Ta gọi C là dãy con chung dài nhất của hai dãy A, B.

Input: Cho trong file COMSEQ.INP:

- Dòng đầu tiên chứa hai số M, N
- Dòng thứ hai chứa M số a_1, a_2, \dots, a_m
- Dòng thứ ba chứa N số b_1, b_2, \dots, b_n .

Output: Ghi ra file COMSEQ.OUT

- Dòng thứ nhất ghi K là số lượng số hạng của dãy C
- Dòng thứ hai ghi K số là các số hạng của dãy C
- Dòng thứ ba ghi K số là chỉ số trong dãy A của K số hạng của C
- Dòng thứ tư ghi K số là chỉ số trong dãy B của K số hạng của C

Ví dụ:

COMSEQ . INP	COMSEQ . OUT
5 6	3
2 3 5 1 4	2 3 5
4 2 3 6 5 10	1 2 3
	2 3 5

Thu gom rác thải**Tên file: garbage.*****

Bản đồ thành phố là một bảng kích thước $M \times N$, các cột được đánh chỉ số từ trái sang phải, các dòng được đánh chỉ số từ trên xuống dưới. Hiện nay thành phố đang bị ô nhiễm nặng, trên mỗi ô (i,j) (giao nhau bởi dòng i và cột j) có $A[i,j]$ đơn vị rác thải.

Thị trường thành phố quyết định sử dụng một con robot đi thu gom rác ở các ô. Robot được lập trình chỉ đi theo hướng xuống dưới, tức là từ ô (i,j) robot chỉ có thể đi tới ô $(i+1,j-1)$, $(i+1,j)$ hoặc $(i+1,j+1)$. Chính vì vậy để robot có thể thu gom được nhiều rác nhất thì người ta phải cho robot xuất phát từ dòng 1 và đi xuống dòng M .

Hãy xác định vị trí cột xuất phát tại dòng 1 và hành trình của robot sao cho robot có thể thu gom được nhiều rác thải nhất.

Dữ liệu vào từ file **GARBAGE.INP**:

- Dòng đầu chứa hai số M và N ($1 \leq M, N \leq 1000$).
- M dòng tiếp theo mỗi dòng ghi N số nguyên $A[i,j]$ ($0 \leq A[i,j] \leq 10^5$) cách nhau bởi 1 dấu cách.

Kết quả ghi ra file **GARBAGE.OUT**:

- Dòng đầu là số lượng rác thải lớn nhất thu được
- Dòng thứ 2 ghi số k là cột mà robot sẽ xuất phát tại dòng 1.
- Dòng thứ $i + 2$ ghi số L là cột mà robot sẽ đi trên hành trình từ dòng 2 tới dòng M

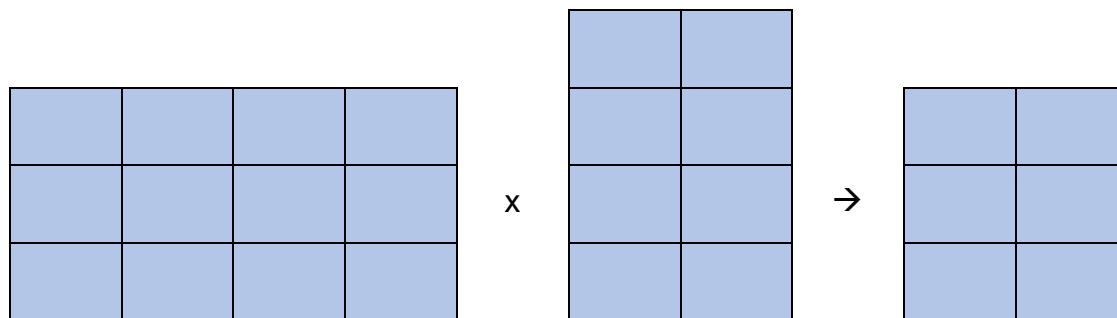
Ví dụ

GARBAGE . INP	GARBAGE . OUT
5 4	26
4 5 2 4	2
3 4 5 2	3
3 4 5 2	3
5 6 3 5	2
4 5 2 5	2

Nhân ma trận

Tên file: **matrix.cpp**

Tích của hai ma trận A kích thước (n, p), ma trận B kích thước (p, m) là ma trận C kích thước (n, m)



Ví dụ trên là tích của hai ma trận kích thước $(3, 4) \times (4, 2) = (3, 2)$

Số phép nhân để thực hiện nhân giữa hai ma trận A(n,p) với B(p,m) được ma trận C(n,m) là $n * p * m$.

Phép nhân ma trận có tính chất kết hợp: $A*B*C = (A*B)*C = A*(B*C)$, thứ tự thực hiện phép nhân được thực hiện trong ngoặc trước, ngoài ngoặc sau.

Cho n ma trận $A_1, A_2, A_3, \dots, A_n$ với kích thước lần lượt là $(d_0, d_1), (d_1, d_2), (d_2, d_3), \dots, (d_{n-1}, d_n)$.

Hãy tìm cách đặt các dấu ngoặc giữa các ma trận để tổng số phép nhân có kết quả là ít nhất.

Input: matrix.inp:

- Dòng 1 chứa số nguyên dương n ($1 \leq n \leq 100$)
- Dòng 2 chứa n + 1 số nguyên dương $d_0, d_1, d_2, \dots, d_n$ là kích thước của các ma trận ($1 \leq d_i \leq 10$).

Output: matrix.out:

- Một dòng duy nhất là số phép nhân thực hiện ít nhất.

Ví dụ:

Matrix.inp	Matrix.out
4 3 4 2 3 4	72

Lucky Luke

Tên file chương trình: **STEP.*****

Bờm chơi trò chơi điện tử Lucky Luke đến màn phải điều khiển Lucky leo lên một cầu thang gồm N bậc ($N \leq 1000$).

Các bậc thang được đánh số từ 1 đến N từ dưới lên trên. Lucky có thể đi lên một bậc thang, hoặc nhảy một bước lên hai bậc thang. Tuy nhiên một số bậc thang đã bị thủng do cũ kỹ và Lucky không thể bước chân lên được. Biết ban đầu, Lucky đứng ở bậc thang số 0 (tức là nền đặt thang).

Chơi đến đây, Bờm chợt nảy ra câu hỏi: có bao nhiêu cách để Lucky leo hết được cầu thang? (nghĩa là leo đến bậc thang thứ N). Bạn hãy giúp Bờm trả lời câu hỏi. Giả thiết rằng Bờm đã biết danh sách các bậc thang bị hỏng.

Dữ liệu: Đọc từ file STEP.INP

- Dòng đầu chứa số N và k là số bậc thang và số lượng bậc thang bị hỏng
- Dòng thứ 2 ghi k số là danh sách của các bậc thang bị thủng.

Kết quả: Ghi ra file STEP.OUT 1 số duy nhất là số dư phép chia của kết quả tìm được cho số 13051984

Ví dụ:

STEP.INP	STEP.OUT
4 1 3	2

Xếp bò

Tên file: **bulls.*****

Nông dân John muốn sắp xếp n con bò (bao gồm cả bò đực và bò cái) của ông ta trên 1 hàng. Ông ta biết rằng những con bò đực rất hung hăng – Nếu 2 con bò đực đứng quá gần nhau thì chúng sẽ trở nên hung dữ và bắt đầu húc nhau, vì thế chúng sẽ phá hỏng mất trật tự trên hàng mà ông ta vừa sắp xếp được.

Theo kinh nghiệm lâu năm của mình, John biết được nếu ở giữa hai con bò đực có ít nhất K con bò cái thì sẽ ngăn chặn được việc chúng húc nhau.

Vì thế, ông ta nhờ các Vcoders giúp đỡ để tính toán xem có bao nhiêu cách sắp xếp đàn bò của ông ta lại sao cho không có “chiến tranh” xảy ra giữa những chú bò, (Tất cả những con bò đực và những con bò cái đều giống nhau). Hai dãy $(B_1..B_n)$ và $(A_1..A_n)$ được cho là khác nhau nếu tồn tại một vị trí i ($1 \leq i \leq n$) sao cho $A_i \neq B_i$.

Input : Đọc từ file **BULLS.INP** gồm 1 dòng duy nhất chứa 2 số N ($1 \leq N \leq 100000$) và k ($0 \leq k \leq n$) cách nhau 1 dấu cách.

Output: Ghi ra file **BULLS.OUT** gồm duy nhất một số là kết quả theo modul 2111992

Ví dụ:

BULLS.INP	BULLS.OUT
4 2	6

Giải thích output(C=bò cái, B:bò đực)

CCCC

BCCC

CBCC

CCBC

CCCB

BCCB

Đổi tiền**Tên file: ATM.CPP**

Một máy rút tiền tự động có n loại tiền mệnh giá lần lượt là d_1, d_2, \dots, d_n đồng. Bạn An cần rút số tiền là S sao cho tổng số tờ tiền rút được là ít nhất có thể.

Em hãy lập trình cho máy ATM có thể đưa ra tổng số lượng tờ tiền thỏa mãn yêu cầu trên. Giả thiết rằng luôn có cách rút tiền tối ưu.

Input: ATM.INP

- Dòng 1: chứa hai số nguyên dương n và S ($1 \leq n \leq 100, 1 \leq S \leq 10000$)
- Dòng 2: chứa n số nguyên dương d_1, d_2, \dots, d_n ($1 \leq d_i \leq 10000$)

Output: ATM.OUT

- Dòng 1: chứa số nguyên K là tổng số tờ tiền ít nhất.
- Dòng 2: chứa n số nguyên k_1, k_2, \dots, k_n là số lượng tờ tiền tương ứng với các mệnh giá d_1, d_2, \dots, d_n .

Ví dụ:

ATM.INP	ATM.OUT
3 35	10
1 2 4	1 1 8