

A - A to Z String 2

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 100 点

問題文

A を N 個、B を N 個、...、Z を N 個この順に繋げて得られる文字列の先頭から X 番目の文字を求めてください。

制約

- $1 \leq N \leq 100$
- $1 \leq X \leq N \times 26$
- 入力は全て整数

入力

入力は以下の形式で標準入力から与えられる。

```
 $N$   $X$ 
```

出力

答えを出力せよ。

入力例 1

```
1 3
```

出力例 1

```
c
```

得られる文字列は ABCDEFGHIJKLMNOPQRSTUVWXYZ です。先頭から 3 番目の文字は c です。

入力例 2

2 12

出力例 2

F

得られる文字列は AABBCDDEEFFGGHHIIJJKLLMMNN00PPQRRSSTTUUVVWXXYYZZ です。先頭から 12 番目の文字は F です。

B - 1D Pawn

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 200 点

問題文

N 個のマスが左右一列に並んでおり、左から順にマス 1、マス 2、...、マス N と番号づけられています。

また、 K 個のコマがあり、最初左から i 番目のコマはマス A_i に置かれています。

これらに対して、 Q 回の操作を行います。 i 回目の操作では次の操作を行います。

- 左から L_i 番目のコマが一番右のマスにあるならば何も行わない。
- そうでない時、左から L_i 番目のコマがあるマスの 1 つ右のマスにコマが無いならば、左から L_i 番目のコマを 1 つ右のマスに移動させる。1 つ右のマスにコマがあるならば、何も行わない。

Q 回の操作が終了した後の状態について、 $i = 1, 2, \dots, K$ に対して左から i 番目のコマがあるマスの番号を出力してください。

制約

- $1 \leq K \leq N \leq 200$
- $1 \leq A_1 < A_2 < \dots < A_K \leq N$
- $1 \leq Q \leq 1000$
- $1 \leq L_i \leq K$
- 入力はすべて整数

入力

入力は以下の形式で標準入力から与えられる。

```
N K Q
A1 A2 ... AK
L1 L2 ... LQ
```

出力

K 個の整数を空白区切りで一行に出力せよ。ここで、 i 個目の整数は、 Q 回の操作が終了した後の状態について、左から i 番目のコマの番号を表す。

入力例 1

```
5 3 5
1 3 4
3 3 1 1 2
```

出力例 1

```
2 4 5
```

最初、コマはマス 1, 3, 4 にあります。これに対して以下のように操作が行われます。

- 左から 3 番目のコマはマス 4 にあります。これは一番右のマスでなく、その 1 つ右のマスにもコマが置かれていないため、左から 3 番目のコマをマス 5 に動かします。コマはマス 1, 3, 5 にある状態になります。
- 左から 3 番目のコマはマス 5 にあります。これは一番右のマスなので、何も行いません。コマはマス 1, 3, 5 にある状態のままです。
- 左から 1 番目のコマはマス 1 にあります。これは一番右のマスでなく、その 1 つ右のマスにもコマが置かれていないため、左から 1 番目のコマをマス 2 に動かします。コマはマス 2, 3, 5 にある状態になります。
- 左から 1 番目のコマはマス 2 にあります。これは一番右のマスではありませんが、その 1 つ右のマス（マス 3）にコマが置かれているため、何も行いません。コマはマス 2, 3, 5 にある状態のままです。
- 左から 2 番目のコマはマス 3 にあります。これは一番右のマスでなく、その右のマスにもコマが置かれていないため、左から 2 番目のコマをマス 4 に動かします。コマはマス 2, 4, 5 にある状態になります。

よって、 Q 回の操作が終わった後でコマはマス 2, 4, 5 に置かれているため、2, 4, 5 を空白区切りでこの順に出力します。

入力例 2

```
2 2 2
1 2
1 2
```

出力例 2

```
1 2
```

入力例 3

```
10 6 9
1 3 5 7 8 9
1 2 3 4 5 6 5 6 2
```

出力例 3

```
2 5 6 7 9 10
```

C - Robot Takahashi

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 300 点

問題文

子供と大人があわせて N 人います。 i 番目の人の体重は W_i です。
それぞれの人が子供か大人かは、 0 と 1 からなる長さ N の文字列 S によって表され、
 S の i 文字目が 0 であるとき i 番目の人が子供であることを、 1 であるとき i 番目の人が大人であることをさします。

ロボットである高橋君に対して実数 X を設定すると、高橋君はそれぞれの人に対して、体重が X 未満なら子供、 X 以上なら大人と判定します。

実数 X に対して $f(X)$ を、高橋君に X を設定したときに N 人のうち子供か大人かを正しく判定できる人数で定めます。

X が実数全体を動くとき、 $f(X)$ の最大値を求めてください。

制約

- $1 \leq N \leq 2 \times 10^5$
- S は 0 と 1 からなる長さ N の文字列
- $1 \leq W_i \leq 10^9$
- N, W_i は整数

入力

入力は以下の形式で標準入力から与えられる。

```
N
S
W_1  W_2  ...  W_N
```

出力

$f(X)$ の最大値を整数で一行に出力せよ。

入力例 1

```
5
10101
60 45 30 40 80
```

出力例 1

```
4
```

$X = 50$ と設定すると、高橋君は 2, 3, 4 番目の人を子供、1, 5 番目の人を大人と判定します。
実際には 2, 4 番目の人が子供、1, 3, 5 番目の人が大人であるので、このとき、1, 2, 4, 5 番目の合計 4 人に対して正しく判定できています。よって、 $f(50) = 4$ です。

5 人全員に対して正しく判定できるような X は存在しないのでこのときが最大です。よって、4 を出力します。

入力例 2

```
3
000
1 2 3
```

出力例 2

```
3
```

例えば、 $X = 10$ とすると最大値 $f(10) = 3$ を達成します。
全員が大人、または全員が子供である可能性もあることに注意してください。

入力例 3

```
5
10101
60 50 50 50 60
```

出力例 3

```
4
```

例えば、 $X = 55$ とすると最大値 $f(55) = 4$ を達成します。
同じ体重の人が複数人存在する可能性もあることに注意してください。

D - Jumping Takahashi 2

実行時間制限: 3 sec / メモリ制限: 1024 MB

配点 : 400 点

問題文

高橋君が住んでいる二次元平面上の街には N 個のジャンプ台があります。 i 番目のジャンプ台は点 (x_i, y_i) にあり、ジャンプ台のパワーは P_i です。また高橋君のジャンプ力は S で表され、はじめ $S = 0$ です。高橋君が訓練を 1 回行う度に S は 1 増えます。

高橋君は以下の条件を満たす場合に限り、 i 番目のジャンプ台から j 番目のジャンプ台にジャンプすることができます。

- $P_i S \geq |x_i - x_j| + |y_i - y_j|$

高橋君の目的は、適切に始点とするジャンプ台を決めることで、そのジャンプ台からどのジャンプ台にも何回かのジャンプで移動できるようにすることです。

目的を達成するためには高橋君は最低で何回訓練を行う必要があるでしょうか？

制約

- $2 \leq N \leq 200$
- $-10^9 \leq x_i, y_i \leq 10^9$
- $1 \leq P_i \leq 10^9$
- $(x_i, y_i) \neq (x_j, y_j) \ (i \neq j)$
- 入力は全て整数

入力

入力は以下の形式で標準入力から与えられる。

```
N
x_1 y_1 P_1
⋮
x_N y_N P_N
```

出力

答えを出力せよ。

入力例 1

```
4
-10 0 1
0 0 5
10 0 1
11 0 1
```

出力例 1

```
2
```

高橋君が2回訓練したとすると、 $S = 2$ です。このとき、2番目のジャンプ台から全てのジャンプ台に移動することができます。

例えば、4番目のジャンプ台へは以下の方法で移動ができます。

- 2番目のジャンプ台から3番目のジャンプ台へジャンプする。（ $P_2S = 10, |x_2 - x_3| + |y_2 - y_3| = 10$ であり、 $P_2S \geq |x_2 - x_3| + |y_2 - y_3|$ を満たす。）
- 3番目のジャンプ台から4番目のジャンプ台へジャンプする。（ $P_3S = 2, |x_3 - x_4| + |y_3 - y_4| = 1$ であり、 $P_3S \geq |x_3 - x_4| + |y_3 - y_4|$ を満たす。）

入力例 2

```
7
20 31 1
13 4 3
-10 -15 2
34 26 5
-2 39 4
0 -50 1
5 -20 2
```

出力例 2

```
18
```

E - Addition and Multiplication 2

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 500 点

問題文

高橋君は整数 x を持っています。最初 $x = 0$ です。

高橋君は以下の操作を好きな回数行えます。

- 整数 i ($1 \leq i \leq 9$) を選ぶ。 C_i 円払い、 x を $10x + i$ で置き換える。

高橋君の予算は N 円です。操作で支払うお金の総和が予算を超過しないように操作を行うとき、最終的に得られる x の最大値を求めてください。

制約

- $1 \leq N \leq 10^6$
- $1 \leq C_i \leq N$
- 入力は全て整数

入力

入力は以下の形式で標準入力から与えられる。

```
N
C_1 C_2 ... C_9
```

出力

答えを出力せよ。

入力例 1

```
5
5 4 3 3 2 5 3 5 3
```

出力例 1

95

例えば $i = 9$ とする操作、 $i = 5$ とする操作を順に行うことで、 x は以下のように変化します。

$0 \rightarrow 9 \rightarrow 95$

操作により支払うお金の合計は $C_9 + C_5 = 3 + 2 = 5$ 円であり、これは予算を超過しません。 予算を超過しないような操作の方法によって 96 以上の整数を作ることが不可能であることが証明できるので、答えは 95 です。

入力例 2

20
1 1 1 1 1 1 1 1 1

出力例 2

99999999999999999999

答えが 64 bit 整数型に収まらないこともあることに注意してください。

F - Teleporter Setting

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 500 点

問題文

N 個の町と M 個のテレポーターがあり、町は町 1, 町 2, ..., 町 N と番号づけられています。それぞれのテレポーターは 2 つの町を双方向に結んでおり、テレポーターを使用する事によってその 2 つの町の間を 1 分で移動することができます。

i 番目のテレポーターは町 U_i と町 V_i を双方向に結んでいますが、いくつかのテレポーターについては結ぶ町の片方が決まっておらず、 $U_i = 0$ のときそのテレポーターが結ぶ町の片方は町 V_i であるが、もう片方が未定であることを意味します。

$i = 1, 2, \dots, N$ それぞれについて、次の問題を解いてください。

結ぶ町の片方が未定となっているテレポーターの結ぶ先をすべて町 i とする。この時に町 1 から町 N まで移動するのに最小で何分かかるか求めよ。町 1 から町 N までテレポーターのみを使って移動するのが不可能な場合は -1 を出力せよ。

制約

- $2 \leq N \leq 3 \times 10^5$
- $1 \leq M \leq 3 \times 10^5$
- $0 \leq U_i < V_i \leq N$
- $i \neq j$ ならば $(U_i, V_i) \neq (U_j, V_j)$
- 入力は全て整数

入力

入力は以下の形式で標準入力から与えられる。

```
N  M
U_1 V_1
U_2 V_2
⋮
U_M V_M
```

出力

N 個の整数を空白区切りで出力せよ。ここで、 k 番目の整数は $i = k$ とした時の問題に対する答えである。

入力例 1

```
3 2
0 2
1 2
```

出力例 1

```
-1 -1 2
```

結ぶ先が未定となっているテレポーターの結び先を町 1 としたとき、
1 番目と 2 番目のテレポーターはともに町 1 と町 2 を結びます。このとき、町 1 から町 3 への移動はできません。

結ぶ先が未定となっているテレポーターの結び先を町 2 としたとき、
1 番目のテレポーターは町 2 同士を、2 番目のテレポーターは町 1 と町 2 を結びます。このときもやはり、町 1 から町 3 への移動はできません。

結ぶ先が未定となっているテレポーターの結び先を町 3 としたとき、
1 番目のテレポーターは町 3 と町 2 を、2 番目のテレポーターは町 1 と町 2 を結びます。
この時、次のようにして町 1 から町 3 へ 2 分で移動できます。

- 2 番目のテレポーターを使用し、町 1 から町 2 まで移動する。
- 1 番目のテレポーターを使用し、町 2 から町 3 まで移動する。

よって、 $-1, -1, 2$ をこの順に出力します。

結ぶ先が未定となっているテレポーターの結び先によっては、同じ町同士を結ぶテレポーターが存在する可能性や、ある 2 つの町を結ぶテレポーターが複数存在する可能性がある事に注意してください。

入力例 2

```
5 5
1 2
1 3
3 4
4 5
0 2
```

出力例 2

3 3 3 3 2

G - Prefix Concatenation

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 600 点

問題文

英小文字のみからなる 2 つの文字列 S, T が与えられます。

(相異なっても良い) S の接頭辞を k 個連結することで T と一致させられるような最小の正整数 k を求めてください。

すなわち、 S の 1 文字目から i 文字目までを取り出した文字列を S_i としたときに、 k 個の 1 以上 $|S|$ 以下の整数の組 (a_1, a_2, \dots, a_k) によって、

$T = S_{a_1} + S_{a_2} + \dots + S_{a_k}$ (ここで $+$ は文字列としての連結を表す) と書くことができるような最小の正整数 k を求めてください。

T と一致させる事が不可能な場合は -1 を出力してください。

制約

- $1 \leq |S| \leq 5 \times 10^5$
- $1 \leq |T| \leq 5 \times 10^5$
- S, T は英小文字のみからなる文字列

入力

入力は以下の形式で標準入力から与えられる。

```
S
T
```

出力

S の接頭辞を k 個連結することで T と一致させられるような最小の正整数 k を出力せよ。 T と一致させる事が不可能な場合は -1 を出力せよ。

入力例 1

```
aba
ababaab
```


出力例 1

3

$T = \text{ababaab}$ は $\text{ab} + \text{aba} + \text{ab}$ と書け、 ab, aba はそれぞれ $S = \text{aba}$ の接頭辞となっています。
 ababaab を 2 個以下の aba の接頭辞の連結によって表す方法はないため、**3** を出力します。

入力例 2

atcoder
ac

出力例 2

-1

T を S の接頭辞の連結によって表す方法はないため、**-1** を出力します。

Ex - Dice Sum 2

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 600 点

問題文

6 面サイコロ専門店「さいころや」には、 N 個のサイコロが売られています。 i 番目のサイコロに書かれている目は $A_{i,1}, A_{i,2}, \dots, A_{i,6}$ であり、価格は C_i です。

高橋君はこの中からちょうど K 個のサイコロを選んで購入します。

現在「さいころや」ではキャンペーンが行われており、購入した K 個のサイコロをそれぞれ一度ずつ振り、出た目の総和の二乗のお金を貰えます。なお、どの目が出るかは一様ランダムであり、各サイコロについて独立です。

買う K 個のサイコロを適切に決めることで、(キャンペーンで貰えるお金) $-$ (購入した K 個のサイコロの価格の合計) の期待値を最大化し、最大化した際の期待値を $\text{mod}998244353$ で求めてください。

▶ 期待値 $\text{mod}998244353$ の定義

制約

- $1 \leq N \leq 1000$
- $1 \leq K \leq N$
- $1 \leq C_i \leq 10^5$
- $1 \leq A_{i,j} \leq 10^5$
- 入力は全て整数

入力

入力は以下の形式で標準入力から与えられる。

```
N K
C1 C2 ... CN
A1,1 A1,2 ... A1,6
⋮
AN,1 AN,2 ... AN,6
```

出力

答えを出力せよ。

入力例 1

```
3 2
1 2 3
1 1 1 1 1 1
2 2 2 2 2 2
3 3 3 3 3 3
```

出力例 1

```
20
```

2 番目のサイコロと 3 番目のサイコロを買うことにすると、(キャンペーンで貰えるお金) − (購入した K 個のサイコロの価格の合計) の期待値は $(2 + 3)^2 - (2 + 3) = 20$ となります。これが期待値の最大値です。

入力例 2

```
10 5
2 5 6 5 2 1 7 9 7 2
5 5 2 4 7 6
2 2 8 7 7 9
8 1 9 6 10 8
8 6 10 3 3 9
1 10 5 8 1 10
7 8 4 8 6 5
1 10 2 5 1 7
7 4 1 4 5 4
5 10 1 5 1 2
5 1 2 3 6 2
```

出力例 2

```
1014
```