

A - ASCII code

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 100 点

問題文

英小文字 a, b, \dots, z の ASCII 文字コードはこの順に $97, 98, \dots, 122$ です。

97 以上 122 以下の整数 N が与えられるので、ASCII 文字コードが N であるような英小文字を出力してください。

制約

- N は 97 以上 122 以下の整数

入力

入力は以下の形式で標準入力から与えられる。

N

出力

答えを出力せよ。

入力例 1

97

出力例 1

a

ASCII 文字コードが 97 である英小文字は a です。

入力例 2

122

出力例 2

z

B - Takahashi's Failure

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 200 点

問題文

高橋君の家には N 個の食品があり、 i 番目の食品のおいしさは A_i です。

また、高橋君には嫌いな食品が K 個あり、具体的には $i = 1, 2, \dots, K$ について、 B_i 番目の食品が嫌いです。

高橋君は N 個の食品のうち、おいしさが最大の食品から 1 つを選んで食べようと考えています。高橋君が嫌いな食品を食べる可能性があるならば Yes を、食べる可能性が無いならば No を出力してください。

制約

- $1 \leq K \leq N \leq 100$
- $1 \leq A_i \leq 100$
- $1 \leq B_i \leq N$
- B_i はすべて相異なる
- 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。

```
 $N$    $K$   
 $A_1$    $A_2$   ...   $A_N$   
 $B_1$    $B_2$   ...   $B_K$ 
```

出力

高橋君が嫌いな食品を食べる可能性があるならば Yes を、無いならば No を出力せよ。

入力例 1

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

出力例 1

Yes

5 個の食品の中でおいしさが最大の食品は食品 3 と 5 の 2 つであり、この 2 つのいずれかを食べます。
高橋君が嫌いな食品は 2, 3, 4 の 3 つであり、そのうち食品 3 を食べる可能性があります。
よって、Yes を出力します。

入力例 2

5 2
100 100 100 1 1
5 4

出力例 2

No

おいしさが最大の食品は食品 1, 2, 3 の 3 つであり、高橋君は嫌いな食品を食べる可能性はありません。

入力例 3

2 1
100 1
2

出力例 3

No

おいしさが最大の食品は食品 1 であり、高橋君は嫌いな食品を食べる可能性はありません。

C - Slot Strategy

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 300 点

問題文

N 個のリールからなるスロットがあります。

i 番目のリールの配列は文字列 S_i によって表されます。ここで、 S_i は $0, 1, \dots, 9$ がちょうど 1 回ずつ現れる長さ 10 の文字列です。

それぞれのリールには対応するボタンがついており、高橋君は各非負整数 t について、スロットが回り始めてからちょうど t 秒後にボタンを 1 つ選んで押す（または何もしない）ことができます。

スロットが回り始めてから t 秒後に i 番目のリールに対応するボタンを押すと、 i 番目のリールは S_i の $(t \bmod 10) + 1$ 文字目を表示して止まります。

ただし、 $t \bmod 10$ で t を 10 で割ったあまりを表します。

高橋君は全てのリールを止めた上で、表示されている文字が全て同じであるようにしたいです。

高橋君が目標を達成できるように全てのリールを止めるまでに、スロットが回り始めてから最小で何秒かかるかを求めてください。

制約

- $2 \leq N \leq 100$
- N は整数
- S_i は $0, 1, \dots, 9$ がちょうど 1 回ずつ現れる長さ 10 の文字列

入力

入力は以下の形式で標準入力から与えられる。

```
N
S_1
S_2
⋮
S_N
```

出力

高橋君が目標を達成できるように全てのリールを止めるまでに、スロットが回り始めてから最小で何秒かかるかを出力せよ。

入力例 1

```
3
1937458062
8124690357
2385760149
```

出力例 1

```
6
```

高橋君は次のようにそれぞれのリールを止めることでスロットが回り始めてから 6 秒後にリールに表示される文字を 8 で揃えることができます。

- スロットの回転開始から 0 秒後に 2 番目のリールに対応するボタンを押します。2 番目のリールは S_2 の $(0 \bmod 10) + 1 = 1$ 文字目である 8 を表示して止まります。
- スロットの回転開始から 2 秒後に 3 番目のリールに対応するボタンを押します。3 番目のリールは S_3 の $(2 \bmod 10) + 1 = 3$ 文字目である 8 を表示して止まります。
- スロットの回転開始から 6 秒後に 1 番目のリールに対応するボタンを押します。1 番目のリールは S_1 の $(6 \bmod 10) + 1 = 7$ 文字目である 8 を表示して止まります。

5 秒以下で全てのリールに表示されている文字を揃える方法はないため、6 を出力します。

入力例 2

```
5
0123456789
0123456789
0123456789
0123456789
0123456789
```

出力例 2

```
40
```

全てのリールを止めた上で、表示されている文字を揃える必要がある事に注意してください。

D - Distinct Trio

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 400 点

問題文

長さ N の数列 $A = (A_1, A_2, \dots, A_N)$ が与えられます。

以下の 2 条件をともに満たすような整数の組 (i, j, k) の個数を求めてください。

- $1 \leq i < j < k \leq N$
- A_i, A_j, A_k は相異なる

制約

- $3 \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 2 \times 10^5$
- 入力に含まれる値は全て整数である

入力

入力は以下の形式で標準入力から与えられる。

```
N
A_1  A_2  ...  A_N
```

出力

答えを出力せよ。

入力例 1

```
4
3 1 4 1
```

出力例 1

```
2
```

条件を満たす整数の組 (i, j, k) は $(1, 2, 3), (1, 3, 4)$ の 2 つです。

入力例 2

10
99999 99998 99997 99996 99995 99994 99993 99992 99991 99990

出力例 2

120

入力例 3

15
3 1 4 1 5 9 2 6 5 3 5 8 9 7 9

出力例 3

355

E - Road Reduction

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 500 点

問題文

AtCoder 王国には都市 $1, 2, \dots, N$ の N 個の都市と、道路 $1, 2, \dots, M$ の M 本の道路があります。道路 i は都市 A_i と B_i を双方向に結び、距離は C_i です。

どの都市間もいくつかの道路を通って行き来することができます。

財政難である王国は、どの都市間もいくつかの道路を通って行き来できるという条件を満たすように $N - 1$ 本の道路を保守し、それ以外の道路を廃道にすることにしました。

保守する道路のみを通って都市 1 から都市 i へ移動するときの距離を d_i とするとき、保守する道路の選び方であって、 $d_2 + d_3 + \dots + d_N$ を最小化するようなものを 1 つ出力してください。

制約

- $2 \leq N \leq 2 \times 10^5$
- $N - 1 \leq M \leq 2 \times 10^5$
- $1 \leq A_i < B_i \leq N$
- $i \neq j$ のとき、 $(A_i, B_i) \neq (A_j, B_j)$
- $1 \leq C_i \leq 10^9$
- どの都市間もいくつかの道路を通って行き来することができる
- 入力に含まれる値は全て整数である

入力

入力は以下の形式で標準入力から与えられる。

```
N M
A1 B1 C1
A2 B2 C2
⋮
AM BM CM
```

出力

保守するような道路の番号を空白区切りで出力せよ。出力の順序は問わない。

答えが複数存在する場合、どれを出力しても正解とみなされる。

入力例 1

```
3 3
1 2 1
2 3 2
1 3 10
```

出力例 1

```
1 2
```

保守する道路の選び方と d_i の値は次のようになります。

- 道路 1, 2 を保守するとき、 $d_2 = 1, d_3 = 3$
- 道路 1, 3 を保守するとき、 $d_2 = 1, d_3 = 10$
- 道路 2, 3 を保守するとき、 $d_2 = 12, d_3 = 10$

よって、道路 1, 2 を保守するときに $d_2 + d_3$ が最小になります。

入力例 2

```
4 6
1 2 1
1 3 1
1 4 1
2 3 1
2 4 1
3 4 1
```

出力例 2

```
3 1 2
```

F - Bread

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 500 点

問題文

長さ L のパンが 1 つあり、これを N 人の子供たちに切り分けます。 i 番目 ($1 \leq i \leq N$) の子供は長さ A_i のパンを欲しがっています。

そこで、高橋君は次の操作を繰り返して、長さ A_1, A_2, \dots, A_N のパンを切り出して配ることにしました。

長さ k のパン 1 つと 1 以上 $k - 1$ 以下の整数 x を選ぶ。選んだパンを長さ x のパンと長さ $k - x$ のパンに切り分ける。
このとき、 x の値によらずコストが k かかる。

それぞれの子供に配るパンは長さがちょうど A_i のパン 1 つである必要がありますが、誰にも配られずに余るパンがいくつあっても構いません。

このとき、全ての子供たちにパンを配るために必要な最小のコストを求めてください。

制約

- $2 \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 10^9$
- $A_1 + A_2 + \dots + A_N \leq L \leq 10^{15}$
- 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  L
A_1 A_2 ... A_N
```

出力

全ての子供たちにパンを配るために必要な最小のコストを出力せよ。

入力例 1

```
5 7
1 2 1 2 1
```

出力例 1

```
16
```

高橋君は次のようにしてパンを切り分けて配ることができます。

- 長さ 7 のパンと整数 $x = 3$ を選ぶ。パンは長さ 3 のパンと長さ 4 のパンに切り分けられる。(コスト 7)
- 長さ 3 のパンと整数 $x = 1$ を選ぶ。パンは長さ 1 のパンと長さ 2 のパンに切り分けられる。前者を 1 番目の子供に配る。(コスト 3)
- 長さ 2 のパンと整数 $x = 1$ を選ぶ。パンは長さ 1 のパン 2 つに切り分けられる。これを 3, 5 番目の子供に配る。(コスト 2)
- 長さ 4 のパンと整数 $x = 2$ を選ぶ。パンは長さ 2 のパン 2 つに切り分けられる。これを 2, 4 番目の子供に配る。(コスト 4)

このとき、コストは $7 + 3 + 2 + 4 = 16$ かかり、これが最小です。また、余るパンはありません。

入力例 2

```
3 1000000000000000000
10000000000 10000000000 10000000000
```

出力例 2

```
1000005000000000000
```

それぞれの子供に配るパンの長さはちょうど A_i でなければならない事に注意してください。

G - Pre-Order

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 600 点

問題文

頂点 1 を根とした N 頂点の根付き木があります。頂点には $1, 2, \dots, N$ の番号がついています。

根から始めて深さ優先探索を行い、行きがけ順で頂点番号を記録したところ、順に P_1, P_2, \dots, P_N となりました。

ただし、深さ優先探索では、現在の頂点に複数の子がある場合、まだ探索していない頂点のうち最も番号が小さい頂点へ移動することとします。

▶ 行きがけ順とは

条件をみたす根付き木として考えられるものの数を **998244353** で割った余りを求めてください。

ただし、ある 2 つの「頂点 1 を根とした N 頂点の根付き木」が異なるとは、ある根以外の頂点が存在して、その親が異なる事を言います。

制約

- $2 \leq N \leq 500$
- $1 \leq P_i \leq N$
- $P_1 = 1$
- P_i はすべて異なる
- 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
P_1 P_2 ... P_N
```

出力

条件をみたす根付き木として考えられるものの数を **998244353** で割った余りを出力せよ。

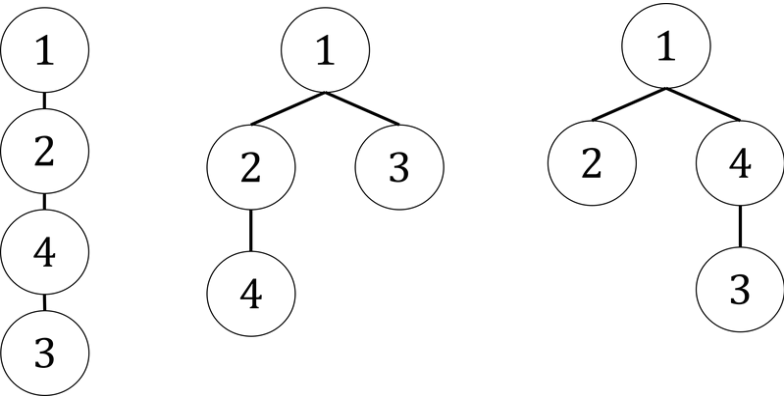
入力例 1

4
1 2 4 3

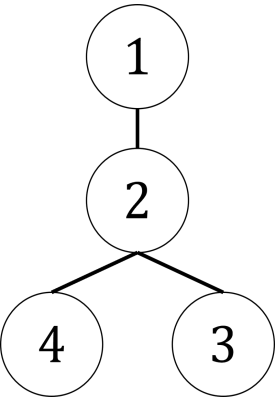
出力例 1

3

条件をみたす根付き木としては次の 3 通りが考えられます。よって、3 を出力します。



また、次のような木は考えられません。頂点 2 の子の頂点のうち、番号の小さい頂点 3 が頂点 4 より先に探索され、このときの行きがけ順は 1, 2, 3, 4 となるからです。



入力例 2

8
1 2 3 5 6 7 8 4

出力例 2

202

Ex - K-th beautiful Necklace

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点 : 600 点

問題文

N 個の宝石があります。 i 番目の宝石は色が D_i で美しさが V_i です。

ここで、色は $1, 2, \dots, C$ のいずれかであり、どの色の宝石も少なくとも 1 個存在します。

N 個の宝石から、色が相異なる C 個の宝石を選んでネックレスを作ります。（選ぶ順番は考えません。）ネックレスの美しさは選んだ宝石の美しさのビットごとの XOR となります。

全てのありえるネックレスの作り方のうち、美しさが K 番目に大きいもののネックレスの美しさを求めてください。（同じ美しさの作り方が複数存在する場合、それらは全て数えます。）

▶ ビットごとの XOR とは

制約

- $1 \leq C \leq N \leq 70$
- $1 \leq D_i \leq C$
- $0 \leq V_i < 2^{60}$
- $1 \leq K \leq 10^{18}$
- 少なくとも K 種類のネックレスを作ることができる
- 入力に含まれる値は全て整数である

入力

入力は以下の形式で標準入力から与えられる。

```
 $N$    $C$    $K$ 
 $D_1$    $V_1$ 
 $\vdots$ 
 $D_N$    $V_N$ 
```

出力

答えを出力せよ。

入力例 1

4 2 3
2 4
2 6
1 2
1 3

出力例 1

5

以下のような 4 種類のネックレスを作ることができます。

- 1, 3 番目の宝石を選ぶ。ネックレスの美しさは $4 \text{ XOR } 2 = 6$ となる。
- 1, 4 番目の宝石を選ぶ。ネックレスの美しさは $4 \text{ XOR } 3 = 7$ となる。
- 2, 3 番目の宝石を選ぶ。ネックレスの美しさは $6 \text{ XOR } 2 = 4$ となる。
- 2, 4 番目の宝石を選ぶ。ネックレスの美しさは $6 \text{ XOR } 3 = 5$ となる。

よって美しさが 3 番目に大きいネックレスの美しさは 5 となります。

入力例 2

3 1 2
1 0
1 0
1 0

出力例 2

0

3 種類のネックレスを作ることができ、いずれも美しさは 0 です。

入力例 3

```
10 3 11
1 414213562373095048
1 732050807568877293
2 236067977499789696
2 449489742783178098
2 645751311064590590
2 828427124746190097
3 162277660168379331
3 316624790355399849
3 464101615137754587
3 605551275463989293
```

出力例 3

```
766842905529259824
```