

A - Five Antennas

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点: 100 点

問題文

AtCoder 市には、5 つのアンテナが一直線上に並んでいます。これらは、西から順にアンテナ A, B, C, D, E と名付けられており、それぞれの座標は a, b, c, d, e です。

2 つのアンテナ間の距離が k 以下であれば直接通信ができ、 k より大きいと直接通信できません。

さて、**直接**通信ができないアンテナの組が存在するかどうか判定してください。

ただし、座標 p と座標 q ($p < q$) のアンテナ間の距離は $q - p$ であるとします。

制約

- a, b, c, d, e, k は 0 以上 123 以下の整数
- $a < b < c < d < e$

入力

入力は以下の形式で標準入力から与えられる。

```
a
b
c
d
e
k
```

出力

直接通信ができないアンテナの組が存在する場合は ':('、存在しない場合は 'Yay!' と出力せよ。

入力例 1

```
1
2
4
8
9
15
```

出力例 1

```
Yay!
```

この場合、直接通信できないアンテナの組は存在しません。なぜなら、

- (A, B) の距離は $2 - 1 = 1$
- (A, C) の距離は $4 - 1 = 3$
- (A, D) の距離は $8 - 1 = 7$
- (A, E) の距離は $9 - 1 = 8$
- (B, C) の距離は $4 - 2 = 2$
- (B, D) の距離は $8 - 2 = 6$
- (B, E) の距離は $9 - 2 = 7$
- (C, D) の距離は $8 - 4 = 4$
- (C, E) の距離は $9 - 4 = 5$
- (D, E) の距離は $9 - 8 = 1$

そのように、全てのアンテナ間の距離が 15 以下であるからです。

よって、'Yay!' と出力すれば正解となります。

入力例 2

```
15
18
26
35
36
18
```

出力例 2

```
:('
```

この場合、アンテナ A と D の距離が $35 - 15 = 20$ となり、18 を超えるため直接通信ができません。

よって、':(' と出力すれば正解となります。

B - Five Dishes

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点: 200 点

問題文

AtCoder 料理店では、以下の 5 つの料理が提供されています。ここで、「調理時間」は料理を注文してから客に届くまでの時間とします。

- ABC 丼：調理時間 A 分
- ARC カレー：調理時間 B 分
- AGC パスタ：調理時間 C 分
- APC ラーメン：調理時間 D 分
- ATC ハンバーグ：調理時間 E 分

また、この店には以下のような「注文のルール」があります。

- 注文は、10 の倍数の時刻 (時刻 0, 10, 20, 30, ...) にしかできない。
- 一回の注文につき一つの料理しか注文できない。
- ある料理を注文したら、それが届くまで別の注文ができない。ただし、料理が届いたちょうどどの時刻には注文ができる。

E869120 君は時刻 0 に料理店に着きました。彼は 5 つの料理全てを注文します。最後の料理が届く最も早い時刻を求めてください。

ただし、料理を注文する順番は自由であり、時刻 0 に注文することも可能とであるとします。

制約

- A, B, C, D, E は 1 以上 123 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。

```
A
B
C
D
E
```

出力

最後の料理が届く最も早い時刻を整数で出力せよ。

入力例 1

```
29
20
7
35
120
```

出力例 1

```
215
```

ABC 丼→ARC カレー→AGC パスタ→ATC ハンバーグ→APC ラーメン の順に注文することになると、各料理の最も早い注文時刻・届く時刻は以下の通りになります。

- 時刻 0 に ABC 丼を注文する。時刻 29 に ABC 丼が届く。
- 時刻 30 に ARC カレーを注文する。時刻 50 に ARC カレーが届く。
- 時刻 50 に AGC パスタを注文する。57 に AGC パスタが届く。
- 時刻 60 に ATC ハンバーグを注文する。時刻 180 に ATC ハンバーグが届く。
- 時刻 180 に APC ラーメンを注文する。時刻 215 に APC ラーメンが届く。

これより早く最後の料理が届くような方法は存在しません。

入力例 2

```
101
86
119
108
57
```

出力例 2

```
481
```

AGC パスタ→ARC カレー→ATC ハンバーグ→APC ラーメン→ABC 丼の順に注文することになると、各料理の最も早い注文時刻・届く時刻は以下の通りになります。

- 時刻 0 に AGC パスタを注文する。時刻 119 に AGC パスタが届く。
- 時刻 120 に ARC カレーを注文する。時刻 206 に ARC カレーが届く。
- 時刻 210 に ATC ハンバーグを注文する。時刻 267 に ATC ハンバーグが届く。
- 時刻 270 に APC ラーメンを注文する。時刻 378 に APC ラーメンが届く。
- 時刻 380 に ABC 丼を注文する。時刻 481 に ABC 丼が届く。

これより早く最後の料理が届くような方法は存在しません。

入力例 3

```
123
123
123
123
123
```

出力例 3

```
643
```

これが入力される最大のケースです。

C - Five Transportations

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点: 300 点

問題文

AtCoder 社は成長し、2028 年になってついに 6 つの都市 (都市 1, 2, 3, 4, 5, 6) からなる AtCoder 帝国を作りました！

AtCoder 帝国には 5 つの交通機関があります。

- 電車：都市 1 から 2 まで 1 分で移動する。1 つの電車には A 人まで乗ることができる。
- バス：都市 2 から 3 まで 1 分で移動する。1 つのバスには B 人まで乗ることができる。
- タクシー：都市 3 から 4 まで 1 分で移動する。1 つのタクシーには C 人まで乗ることができる。
- 飛行機：都市 4 から 5 まで 1 分で移動する。1 つの飛行機には D 人まで乗ることができる。
- 船：都市 5 から 6 までを 1 分で移動する。1 つの船には E 人まで乗ることができる。

それぞれの交通機関は、各整数時刻 $(0, 1, 2, 3, \dots)$ に、都市から出発します。

いま、 N 人のグループが都市 1 におり、全員都市 6 まで移動したいです。全員が都市 6 に到着するまでに最短で何分かかるでしょうか？

なお、乗り継ぎにかかる時間を考える必要はありません。

制約

- $1 \leq N, A, B, C, D, E \leq 10^{15}$
- 入力中の値はすべて整数である。

入力

入力は以下の形式で標準入力から与えられる。

```
N
A
B
C
D
E
```

出力

全員が都市 6 に移動するのに必要な最小の時間を分単位で出力せよ。

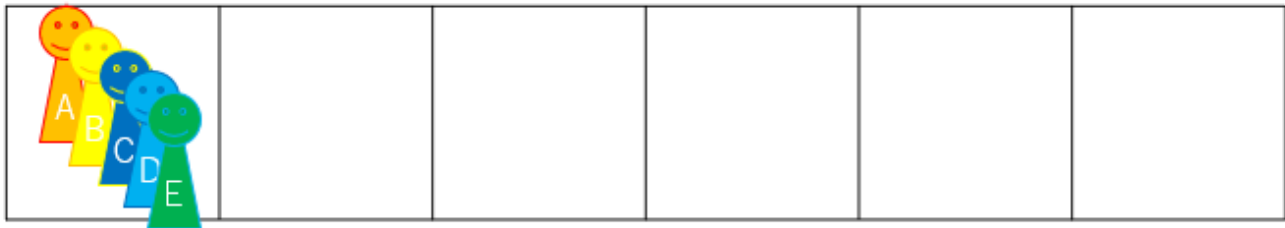
入力例 1

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

出力例 1

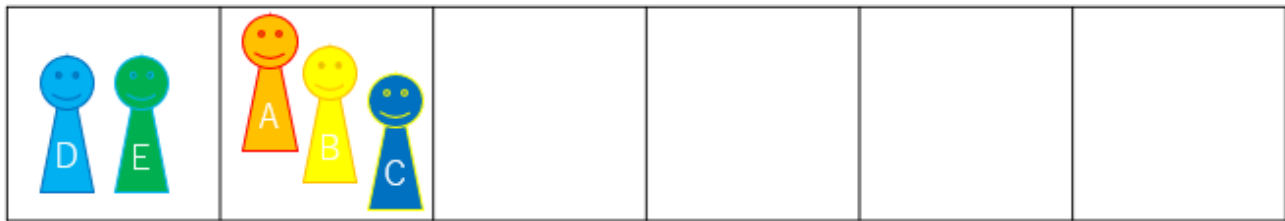
7

例えば、次のような移動方法が考えられます。
はじめ、次の画像のように、 $N = 5$ 人が都市 1 にいます。



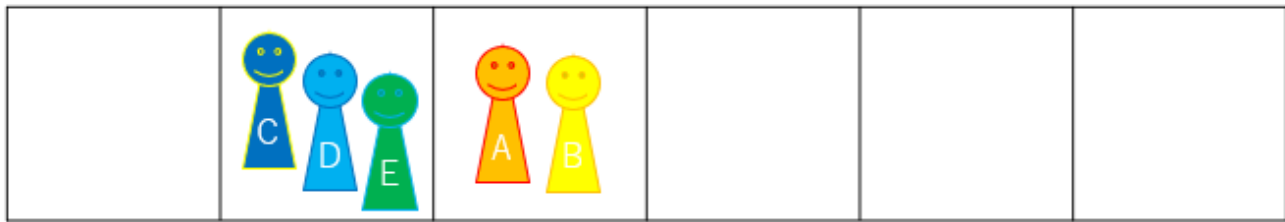
TIME: 0

1 分後までに、3 人が都市 1 から都市 2 に電車で移動します。ここで、電車は一度に 3 人までしか運べないことに注意してください。



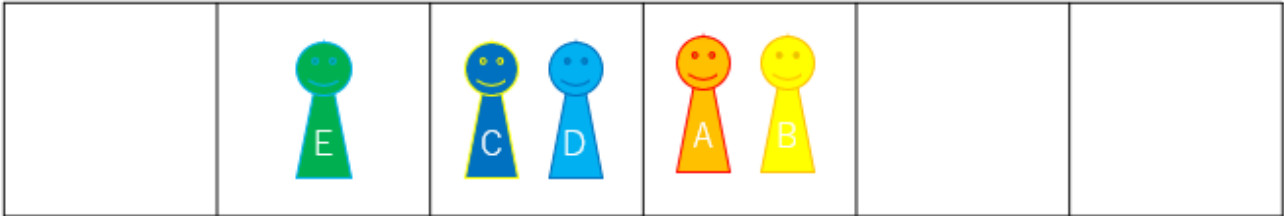
TIME: 1

2 分後までに、残り 2 人が都市 1 から都市 2 に電車で移動し、都市 2 にいた 3 人のうち 2 人がバスで都市 3 に移動します。ここで、バスは一度に 2 人までしか運べないことに注意してください。



TIME: 2

3 分後までに、2 人が都市 2 から都市 3 にバスで移動し、2 人が都市 3 から都市 4 にタクシーで移動します。



TIME: 3

それ以降は、まだ都市 6 に到着していない人が止まらずに移動し続けると、全員が 7 分で都市 6 に着くことができます。

また、6 分以内で全員が都市 6 に着く方法はありません。

入力例 2

```
10
123
123
123
123
123
```

出力例 2

```
5
```

どの交通機関も $N = 10$ 人を 1 回で運ぶことができます。

したがって、全員が止まらずに移動し続ければ 5 分で都市 6 に着くことができます。

入力例 3

```
10000000007
2
3
5
7
11
```

出力例 3

```
50000000008
```

入力・出力が 32 ビット整数型に収まらない可能性があることに注意してください。

D - Cake 123

実行時間制限: 2 sec / メモリ制限: 1024 MB

配点: 400 点

問題文

AtCoder 洋菓子店は数字の形をしたキャンドルがついたケーキを販売しています。

ここには 1, 2, 3 の形をしたキャンドルがついたケーキがそれぞれ X 種類、 Y 種類、 Z 種類あります。それぞれのケーキには「美味しさ」という整数の値が以下のように割り当てられています。

- 1 の形のキャンドルがついたケーキの美味しさはそれぞれ A_1, A_2, \dots, A_X
- 2 の形のキャンドルがついたケーキの美味しさはそれぞれ B_1, B_2, \dots, B_Y
- 3 の形のキャンドルがついたケーキの美味しさはそれぞれ C_1, C_2, \dots, C_Z

高橋君は ABC 123 を記念するために、1, 2, 3 の形のキャンドルがついたケーキを 1 つずつ買うことにしました。

そのようにケーキを買う方法は $X \times Y \times Z$ 通りあります。

これらの選び方を 3 つのケーキの美味しさの合計が大きい順に並べたとき、 $1, 2, \dots, K$ 番目の選び方でのケーキの美味しさの合計をそれぞれ出力してください。

制約

- $1 \leq X \leq 1\,000$
- $1 \leq Y \leq 1\,000$
- $1 \leq Z \leq 1\,000$
- $1 \leq K \leq \min(3\,000, X \times Y \times Z)$
- $1 \leq A_i \leq 10\,000\,000\,000$
- $1 \leq B_i \leq 10\,000\,000\,000$
- $1 \leq C_i \leq 10\,000\,000\,000$
- 入力中の値はすべて整数である。

入力

入力は以下の形式で標準入力から与えられる。

```
X Y Z K
A1 A2 A3 ... AX
B1 B2 B3 ... BY
C1 C2 C3 ... CZ
```

出力

i 行目に、問題文中の i 番目の値を出力せよ。

入力例 1

```
2 2 2 8
4 6
1 5
3 8
```

出力例 1

```
19
17
15
14
13
12
10
8
```

3つのケーキの選び方は $2 \times 2 \times 2 = 8$ 通りあり、それらをケーキの美味しさの合計が大きい順に並べると以下の通りです。

- $(A_2, B_2, C_2): 6 + 5 + 8 = 19$
- $(A_1, B_2, C_2): 4 + 5 + 8 = 17$
- $(A_2, B_1, C_2): 6 + 1 + 8 = 15$
- $(A_2, B_2, C_1): 6 + 5 + 3 = 14$
- $(A_1, B_1, C_2): 4 + 1 + 8 = 13$
- $(A_1, B_2, C_1): 4 + 5 + 3 = 12$
- $(A_2, B_1, C_1): 6 + 1 + 3 = 10$
- $(A_1, B_1, C_1): 4 + 1 + 3 = 8$

入力例 2

```
3 3 3 5
1 10 100
2 20 200
1 10 100
```

出力例 2

```
400
310
310
301
301
```

美味しさの合計が同じになる組み合わせが複数ある可能性があります。例えば、このテストケースで (A_1, B_3, C_3) を選ぶときと (A_3, B_3, C_1) を選ぶときはともに、美味しさの合計が 301 となります。しかし、これらは異なる選び方であるため、出力には 301 が 2 回出現します。

入力例 3

```
10 10 10 20
7467038376 5724769290 292794712 2843504496 3381970101 8402252870 249131806 6310293640 6690322794 6082
257488
1873977926 2576529623 1144842195 1379118507 6003234687 4925540914 3902539811 3326692703 484657758 287
7436338
4975681328 8974383988 2882263257 7690203955 514305523 6679823484 4263279310 585966808 3752282379 6205
85736
```

出力例 3

```
23379871545
22444657051
22302177772
22095691512
21667941469
21366963278
21287912315
21279176669
21160477018
21085311041
21059876163
21017997739
20703329561
20702387965
20590247696
20383761436
20343962175
20254073196
20210218542
20150096547
```

入力・出力は 32 ビット整数に収まらない可能性があることに注意してください。