

第19回 日本情報オリンピック本選 課題

2020年2月9日

競技参加者への注意事項

- 競技開始の指示のあるまで、課題が入った封筒を開けないこと。
- 課題は5問である。競技時間は4時間（9時～13時）である。
- 質問がある場合は、競技開始から2時間経過するまで質問票を提出することができる。質問への回答は「正しい」「正しくない」「ノーコメント」「課題の記述の中に答えはある」「無効」のいずれかである。競技開始から2時間経過後も質問票を提出することはできるが、競技時間内に回答するとは限らない。
- 実行時間制限・メモリ制限・コンパイラオプション・配点は Overview Sheet に記載する。
- すべての課題でセットごとに採点を行う。各セットは1個もしくは複数のデータで構成されており、セット中のすべてのデータに正解した場合にのみ配点分の得点が与えられる。
- 各課題がいくつかの小課題に分割されることがある。詳細は問題文中に記載する。
- 解答は、競技サーバ <http://joi/> から提出すること。
- 競技終了15分前までに提出されたソースには競技時間内にフィードバックが与えられる。それ以降に提出されたソースには競技時間内にフィードバックが与えられるとは限らない。フィードバックが与えられるまでに時間がかかることがある。
- 各課題ごとに、最後の提出から1分間は解答を提出することができない。各課題ごとに、ソースを提出することができる回数は50回までである。
- ソースを複数回提出した場合は、提出された全てのソースが採点される。各小課題に対し、提出された全てのソースにおけるその小課題の得点の最大値が、その小課題の最終的な得点となる。各課題の得点は、その課題に含まれる小課題の最終的な得点の合計である。
- 提出した解答ファイルは手元に保存しておくこと。競技実施中に競技サーバに予期せぬ障害が発生した場合に、解答ファイルの再提出が必要になる場合がある。

情報オリンピック日本委員会



この課題はクリエイティブ・コモンズ表示-継承4.0国際ライセンス (CC BY-SA 4.0) の下で利用可能です。クリエイティブ・コモンズ・ライセンスの詳細は、クリエイティブ・コモンズ・ジャパンのウェブページ (<http://creativecommons.jp/>) をご覧ください。表示-継承4.0国際 (CC BY-SA 4.0) 以外のライセンスを希望される場合は、情報オリンピック日本委員会までご連絡ください。



1

長いだけのネクタイ (Just Long Neckties)

あなたは Just Odd Inventions 社を知っているだろうか？この会社の業務は「ただ奇妙な発明 (just odd inventions)」をすることである。ここでは略して JOI 社と呼ぶ。

JOI 社は新商品「長いだけのネクタイ」を開発した。ネクタイは $N+1$ 種類あり、各種類には 1 から $N+1$ までの番号がついている。 i 番目 ($1 \leq i \leq N+1$) の種類のネクタイの長さは A_i である。

JOI 社は社員を集め、ネクタイの試着会を行うことにした。試着会には N 人の社員が参加し、 j 人目 ($1 \leq j \leq N$) の社員がはじめに付けているネクタイの長さは B_j である。

試着会は以下の手順で行われる予定である。

1. まず、試着会で使わないネクタイを 1 種類選ぶ。
2. 次に、各社員はそれ以外のネクタイから試着するネクタイを 1 種類選ぶ。ただし、どの 2 人も同じ種類のネクタイを選ばないようにする。
3. 最後に、各社員は今付けているネクタイを外し、先ほど選んだネクタイを試着する。

長さ b のネクタイを付けていた社員が、長さ a のネクタイを試着すると大きさ $\max\{a-b, 0\}$ の奇妙さを感じる。(ここで、 $\max\{a-b, 0\}$ は、 $a-b$ と 0 のうち小さくない方を表す。) 試着会において各社員の感じる奇妙さの最大値を、その試着会の奇妙さとする。

試着会で使わないネクタイが k 番目の種類のネクタイのとき、試着会の奇妙さとして考えられる最小の値を C_k とする。

各種類のネクタイの長さ、各社員がはじめに付けているネクタイの長さが与えられたとき、 C_1, C_2, \dots, C_{N+1} の値を求めるプログラムを作成せよ。

入力

入力は以下の形式で標準入力から与えられる。入力される値はすべて整数である。

N
 $A_1 \dots A_{N+1}$
 $B_1 \dots B_N$

出力

C_1, C_2, \dots, C_{N+1} の値を、空白区切りで標準出力に 1 行で出力せよ。



制約

- $1 \leq N \leq 200\,000$.
- $1 \leq A_i \leq 1\,000\,000\,000$ ($1 \leq i \leq N + 1$).
- $1 \leq B_j \leq 1\,000\,000\,000$ ($1 \leq j \leq N$).

小課題

1. (1 点) $N \leq 10$.
2. (8 点) $N \leq 2\,000$.
3. (91 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
3 4 3 7 6 2 6 4	2 2 1 1

例えば、試着会は次のように行われる.

- 4 番目の種類のネクタイを使わないことにする.
- 社員 1 が 1 番目の, 社員 2 が 2 番目の, 社員 3 が 3 番目の種類のネクタイを選ぶ.
- 各社員が試着する.

このとき, 各社員が感じる奇妙さは順に 2, 0, 3 となるから, この試着会の奇妙さは 3 である.

社員が選ぶネクタイを変えることで, 試着会の奇妙さを 1 にすることができる. 例えば, 試着会を次のように行うとする.

- 4 番目の種類のネクタイを使わないことにする.
- 社員 1 が 2 番目の, 社員 2 が 3 番目の, 社員 3 が 1 番目の種類のネクタイを選ぶ.
- 各社員が試着する.

このとき, 各社員が感じる奇妙さは順に 1, 1, 0 となるから, この試着会の奇妙さは 1 である.

これが 4 番目の種類のネクタイを使わない場合の試着会の奇妙さの最小値なので, $C_4 = 1$ である.



入力例 2	出力例 2
5 4 7 9 10 11 12 3 5 7 9 11	4 4 3 2 2 2



2

JJOII 2 (JJOII 2)

ビ太郎は友人のビバ子から誕生日プレゼントに J, O, I の 3 種類の文字からなる長さ N の文字列 S をもらった。

K を 1 以上の整数とする。 K 個の文字 J, K 個の文字 O, K 個の文字 I をこの順に並べた文字列をレベル K の JOI 文字列と呼ぶことにする。例えば, JJOOII はレベル 2 の JOI 文字列である。

ビ太郎はレベル K の JOI 文字列が好きなので, 以下の 3 種類の操作を任意の回数, 任意の順番で行うことで, 文字列 S をレベル K の JOI 文字列に変換することにした。

操作 1 文字列 S の先頭の文字を消す。

操作 2 文字列 S の末尾の文字を消す。

操作 3 文字列 S の先頭でも末尾でもない文字を消す。

操作 3 を行うのは面倒なので, 操作 3 を行う回数をできるだけ少なくして, 文字列 S をレベル K の JOI 文字列に変換したい。

長さ N の文字列 S と 1 以上の整数 K が与えられたとき, 文字列 S をレベル K の JOI 文字列に変換するのに必要な操作 3 の回数の最小値を出力するプログラムを作成せよ。ただし, どのように操作を行っても文字列 S をレベル K の JOI 文字列に変換できない場合は, 代わりに -1 を出力せよ。

入力

入力は以下の形式で標準入力から与えられる。 N, K は整数である。 S は文字列である。

$N K$

S

出力

文字列 S をレベル K の JOI 文字列に変換するのに必要な操作 3 の回数の最小値を 1 行で出力せよ。ただし, どのように操作を行っても文字列 S をレベル K の JOI 文字列に変換できない場合は, 代わりに -1 を出力せよ。

制約

- $3 \leq N \leq 200\,000$.
- $1 \leq K \leq \frac{N}{3}$.
- S は J, O, I の 3 種類の文字からなる長さ N の文字列である。



小課題

1. (1 点) $N \leq 21$.
2. (12 点) $N \leq 3000$.
3. (87 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
10 2 OJIJOIOIJJ	2

次のように操作を行うことで、文字列 S をレベル K の JOI 文字列に変換できる.

1. まず操作 1 を行う. 文字列 S は JIJJOIOIJJ になる.
 2. 次に操作 2 を行う. 文字列 S は JIJJOIOII になる.
 3. 次に操作 3 を行い, 先頭から 2 文字目を消す. 文字列 S は JJJOIOII になる.
 4. 最後に操作 3 を行い, 先頭から 4 文字目を消す. 文字列 S は JJOOII になる.
- 2 回未満の操作 3 で変換することは不可能なので, 2 を出力する.

入力例 2	出力例 2
9 3 JJJ000III	0

操作を行わなくてもよい.

入力例 3	出力例 3
9 1 III000JJJ	-1

この入力例では, どのように操作を行って文字列 S をレベル 1 の JOI 文字列に変換できない.



3

スタンプラリー 3 (Collecting Stamps 3)

JOI 君が住む IOI 国は、大きな湖があることで有名である。今日、湖の周りでスタンプラリー大会が行われることになった。

湖の周りには N 個のスタンプ台が設置されており、時計回りに 1 から N までの番号が付いている。湖の周りの長さは L メートルであり、スタンプ台 i ($1 \leq i \leq N$) はスタンプラリーのスタート地点から湖の周りに沿って時計回りに X_i メートルだけ進んだ地点に設置されている。

スタンプラリーの各参加者は、スタンプラリー開始時にはスタート地点にいて、スタンプラリー開始後は湖の周りに沿って時計回りもしくは反時計回りに移動することができる。参加者は、スタンプ台が設置されている地点に到着したとき、まだそのスタンプ台でスタンプを押していなかった場合に限り、スタンプを 1 回だけ押すことができる。ただし、スタンプ台 i ($1 \leq i \leq N$) はスタンプラリー開始から T_i 秒が経過すると撤去され、それより後に参加者が到着してもそのスタンプ台でスタンプを押すことはできなくなる。なお、 T_i 秒ちょうどに参加者が到着した場合については、スタンプを押すことができるとする。

JOI 君はこのスタンプラリー大会の参加者である。JOI 君は 1 メートルを進むのに 1 秒かかる。また、JOI 君はスタンプを押すことに熟練しているので、スタンプを押すのにかかる時間は無視することができる。

スタンプ台の個数、湖の周りの長さ、各スタンプ台が設置されている地点、各スタンプ台が撤去される時刻が与えられたとき、JOI 君が押すことのできるスタンプの個数の最大値を求めるプログラムを作成せよ。

入力

入力は以下の形式で標準入力から与えられる。入力される値はすべて整数である。

$N L$

$X_1 \dots X_N$

$T_1 \dots T_N$

出力

JOI 君が押すことのできるスタンプの個数の最大値を、標準出力に 1 行で出力せよ。

制約

- $1 \leq N \leq 200$.
- $2 \leq L \leq 1\,000\,000\,000$.
- $1 \leq X_i < L$ ($1 \leq i \leq N$).



- $X_i < X_{i+1}$ ($1 \leq i \leq N - 1$).
- $0 \leq T_i \leq 1\,000\,000\,000$ ($1 \leq i \leq N$).

小課題

1. (5 点) $N \leq 12$, $L \leq 200$, $T_i \leq 200$ ($1 \leq i \leq N$).
2. (10 点) $N \leq 15$.
3. (10 点) $L \leq 200$, $T_i \leq 200$ ($1 \leq i \leq N$).
4. (75 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
6 25 3 4 7 17 21 23 11 7 17 10 8 10	4

以下のようにすると JOI 君は 4 個のスタンプを押すことができる.

1. 反時計回りに 2 メートル進む. スタンプラリー開始からの経過時間は 2 秒であるので, スタンプ台 6 でスタンプを押すことができる.
2. さらに反時計回りに 2 メートル進む. スタンプラリー開始からの経過時間は 4 秒であるので, スタンプ台 5 でスタンプを押すことができる.
3. 時計回りに 7 メートル進む. スタンプラリー開始からの経過時間は 11 秒であるので, スタンプ台 1 でスタンプを押すことができる.
4. さらに時計回りに 1 メートル進む. スタンプラリー開始からの経過時間は 12 秒であるので, スタンプ台 2 でスタンプを押すことはできない.
5. さらに時計回りに 3 メートル進む. スタンプラリー開始からの経過時間は 15 秒であるので, スタンプ台 3 でスタンプを押すことができる.

どのように移動しても JOI 君が 5 個以上のスタンプを押すことはできないので, 4 を出力する.

入力例 2	出力例 2
5 20 4 5 8 13 17 18 23 15 7 10	5



JOI 君はスタンプラリー開始後、湖の周りを反時計回りに進み続けることで、すべてのスタンプ台でスタンプを押すことができる。

入力例 3	出力例 3
4 19 3 7 12 14 2 0 5 4	0

残念ながら、JOI 君がどのように移動したとしてもスタンプを押すことはできない。

入力例 4	出力例 4
10 87 9 23 33 38 42 44 45 62 67 78 15 91 7 27 31 53 12 91 89 46	5



4

オリンピックバス (Olympic Bus)

JOI 国には N 個の都市があり、1 から N までの番号が付いている。また、都市と都市を一方方向に結ぶ M 本のバス路線があり、1 から M までの番号が付いている。バス路線 i ($1 \leq i \leq M$) は都市 U_i から都市 V_i へ向けて運行されており、運賃は C_i 円である。バス路線 i ($1 \leq i \leq M$) では、都市 U_i 以外で乗ったり、都市 V_i 以外で降りることはできない。ある都市からある都市へ向けて運行されるバス路線が複数存在するかもしれない。

JOI 国では間もなくオリンピックが開催される。JOI 国の運輸大臣である K 理事長は、バス路線を高々 1 つ選び、オリンピック期間中、運賃を変更せずにそのバス路線の運行方向を反転させることにした。つまり、バス路線 i ($1 \leq i \leq M$) を選んだ場合、オリンピック期間中、そのバス路線は都市 U_i から都市 V_i へ向けて運行されるのではなく、都市 V_i から都市 U_i へ向けて運行されるようになる。ただし、バス路線 i の運行方向の反転には D_i 円かかり、これは K 理事長のポケットマネーにより賄われる。また、混乱を避けるため、オリンピック期間の途中でバス路線を反転させることはできない。

運輸大臣である K 理事長は、オリンピック期間中、都市 1 と都市 N の間をバス路線を乗り継いで往復する予定である。運行方向を反転させるバス路線をうまく選ぶことで、往復の合計運賃と運行方向の反転の費用との和を最小化したい。

都市の個数と、バス路線の情報が与えられたとき、運行方向を反転させるバス路線をうまく選ぶことで、都市 1 と都市 N の間の往復の合計運賃と、運行方向の反転の費用との和の最小値を求めるプログラムを作成せよ。ただし、どのようにバス路線を選んでも都市 1 と都市 N の間を往復することができない場合は、代わりに -1 を出力せよ。

入力

入力は以下の形式で標準入力から与えられる。入力される値はすべて整数である。

```
N M
U1 V1 C1 D1
⋮
UM VM CM DM
```

出力

都市 1 と都市 N の間の往復の合計運賃と、運行方向の反転の費用との和の最小値を、標準出力に 1 行で出力せよ。ただし、どのようにバス路線を選んでも都市 1 と都市 N の間を往復することができない場合は、代わりに -1 を出力せよ。



制約

- $2 \leq N \leq 200$.
- $1 \leq M \leq 50\,000$.
- $1 \leq U_i \leq N$ ($1 \leq i \leq M$).
- $1 \leq V_i \leq N$ ($1 \leq i \leq M$).
- $U_i \neq V_i$ ($1 \leq i \leq M$).
- $0 \leq C_i \leq 1\,000\,000$ ($1 \leq i \leq M$).
- $0 \leq D_i \leq 1\,000\,000\,000$ ($1 \leq i \leq M$).

小課題

1. (5 点) $M \leq 1000$.
2. (11 点) M は偶数, $U_{2i-1} = U_{2i}$, $V_{2i-1} = V_{2i}$, $C_{2i-1} = C_{2i}$ ($1 \leq i \leq \frac{M}{2}$).
3. (21 点) $C_i = 0$ ($1 \leq i \leq M$).
4. (63 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
4 5 1 2 4 4 1 3 2 1 4 3 1 2 4 1 6 1 2 4 2 5	10

バス路線 2 の運行方向を費用 1 で反転させると, 都市 1 から都市 4 への移動にかかる運賃は最小で 6, 都市 4 から都市 1 への移動にかかる運賃は最小で 3 となり, 都市 1 と都市 4 の間の往復の合計運賃と, 運行方向の反転の費用との和は 10 となる.

都市 1 と都市 4 の間の往復の合計運賃と, 運行方向の反転の費用との和を 10 より小さくすることはできないので, 10 を出力する.



入力例 2	出力例 2
4 10 1 2 4 4 1 2 4 4 1 3 2 1 1 3 2 1 4 3 1 2 4 3 1 2 4 1 6 1 4 1 6 1 2 4 2 5 2 4 2 5	10

この入出力例は小課題 2 の制約を満たす。

入力例 3	出力例 3
4 4 1 2 0 4 1 3 0 1 4 3 0 2 4 1 0 1	2

この入出力例は小課題 3 の制約を満たす。

入力例 4	出力例 4
4 5 1 2 4 4 1 3 2 4 4 3 1 5 4 1 6 1 2 4 2 5	12

バス路線の運行方向を反転させなくてもよい。

入力例 5	出力例 5
4 5 2 1 4 4 1 3 2 1 4 3 1 2 4 3 6 1 2 4 2 5	-1

この入力例では、都市 4 から都市 3 へと運行されるバス路線が 2 本存在する。



5

火事 (Fire)

JOI 村には N 個の区画があり、1 から N までの番号が付いている。これらの区画は番号順に一直列に並んでいる。今、各区画では火事が発生しており、時刻 0 における区画 i ($1 \leq i \leq N$) の火の強さは S_i ($S_i > 0$) である。

時刻 0 に、区画 1 から区画 N の方向に風が吹き始めた。隣り合う 2 つの区画について、時刻 t ($0 \leq t$) において風上の区画の火が風下の区画の火より強いとき、時刻 $t+1$ における風下の区画の火の強さは、時刻 t における風上の区画の火の強さと同じになってしまう。そうでないときは、時刻 $t+1$ における風下の区画の火の強さは、時刻 t と同じである。すなわち、時刻 t ($0 \leq t$) における区画 i ($1 \leq i \leq N$) の火の強さを $S_i(t)$ と書くとする、 $1 \leq t$ ならば、 $S_i(t) = \max\{S_{i-1}(t-1), S_i(t-1)\}$ となる。ただし、任意の t ($0 \leq t$) に対して、 $S_0(t) = 0$ とし、任意の i ($1 \leq i \leq N$) に対し $S_i(0) = S_i$ とする。

消防士であるあなたは Q 個の消火活動を計画した。 Q 個の計画のうちどれか 1 つだけを実施する予定である。 j 番目の計画 ($1 \leq j \leq Q$) は、時刻 T_j に、 $L_j \leq k \leq R_j$ となるすべての区画 k に消火剤を撒き、それらの区画を消火するというものである。火の強さが s である区画を消火するためには s リットルの消火剤が必要である。つまり、 j 番目の計画の消火活動には $S_{L_j}(T_j) + S_{L_j+1}(T_j) + \dots + S_{R_j}(T_j)$ リットルの消火剤が必要である。

どの計画を実行するか吟味するためにも、各計画に必要な消火剤の量が知りたい。

時刻 0 における火の強さの情報と消火活動の計画の情報が与えられたとき、各計画に必要な消火剤の量を求めるプログラムを作成せよ。

入力

入力は以下の形式で標準入力から与えられる。入力される値はすべて整数である。

N Q
 $S_1 \dots S_N$
 T_1 L_1 R_1
⋮
 T_Q L_Q R_Q

出力

標準出力に Q 行で出力せよ。第 j 行目 ($1 \leq j \leq Q$) には j 番目の計画に必要な消火剤の量を出力せよ。



制約

- $1 \leq N \leq 200\,000$.
- $1 \leq Q \leq 200\,000$.
- $1 \leq S_i \leq 1\,000\,000\,000$ ($1 \leq i \leq N$).
- $1 \leq T_j \leq N$ ($1 \leq j \leq Q$).
- $1 \leq L_j \leq R_j \leq N$ ($1 \leq j \leq Q$).

小課題

1. (1 点) $N \leq 200$, $Q \leq 200$.
2. (6 点) $T_1 = T_2 = \dots = T_Q$.
3. (7 点) $L_j = R_j$ ($1 \leq j \leq Q$).
4. (6 点) $S_i \leq 2$ ($1 \leq i \leq N$).
5. (80 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
5 5	21
9 3 2 6 5	39
1 1 3	33
2 1 5	9
3 2 5	27
4 3 3	
5 3 5	

- 時刻 0 における各区画の火の強さは、区画 1 から順に 9, 3, 2, 6, 5 である。
- 時刻 1 における各区画の火の強さは、区画 1 から順に 9, 9, 3, 6, 6 である。よって、1 番目の計画に必要な消火剤の量は $9 + 9 + 3 = 21$ リットルである。
- 時刻 2 における各区画の火の強さは、区画 1 から順に 9, 9, 9, 6, 6 である。よって、2 番目の計画に必要な消火剤の量は $9 + 9 + 9 + 6 + 6 = 39$ リットルである。



- 時刻 3 における各区画の火の強さは、区画 1 から順に 9,9,9,9,6 である。よって、3 番目の計画に必要な消火剤の量は $9+9+9+6=33$ リットルである。
- 時刻 4 における各区画の火の強さは、区画 1 から順に 9,9,9,9,9 である。よって、4 番目の計画に必要な消火剤の量は 9 リットルである。
- 時刻 5 における各区画の火の強さは、区画 1 から順に 9,9,9,9,9 である。よって、5 番目の計画に必要な消火剤の量は $9+9+9=27$ リットルである。

入力例 1 は小課題 1, 5 の制約を満たす。

入力例 2	出力例 2
10 10	28
3 1 4 1 5 9 2 6 5 3	21
1 1 6	34
2 8 10	4
4 2 7	64
8 3 3	43
6 1 10	55
3 2 8	9
5 1 9	27
7 4 5	9
9 7 9	
10 10 10	

入力例 2 は小課題 1, 5 の制約を満たす。

入力例 3	出力例 3
10 10	9
3 1 4 1 5 9 2 6 5 3	9
1 6 6	3
2 8 8	4
4 2 2	3
8 3 3	4
6 1 1	5
3 4 4	9
5 5 5	9
7 10 10	9
9 8 8	
10 7 7	

入力例 3 は小課題 1, 3, 5 の制約を満たす。



入力例 4	出力例 4
10 10	28
3 1 4 1 5 9 2 6 5 3	27
7 1 6	34
7 8 10	4
7 2 7	64
7 3 3	43
7 1 10	55
7 2 8	9
7 1 9	27
7 4 5	9
7 7 9	
7 10 10	

入力例 4 は小課題 1, 2, 5 の制約を満たす。

入力例 5	出力例 5
20 20	25
2 1 2 2 1 1 1 1 2 2 2 1 2 1 1 2 1 2 1 1	30
1 1 14	12
2 3 18	32
4 10 15	2
8 2 17	24
9 20 20	38
4 8 19	10
7 2 20	14
11 1 5	40
13 2 8	8
20 1 20	28
2 12 15	24
7 1 14	32
12 7 18	4
14 2 17	2
9 19 20	28
12 12 12	28
6 2 15	12
11 2 15	40
19 12 17	
4 1 20	

入力例 5 は小課題 1, 4, 5 の制約を満たす。