



リンゴの出荷 (Apples)

JOI 農場ではリンゴの入荷と出荷を行っている。JOI 農場は 1 つずつリンゴを入荷し、出荷依頼が来た場合は次の入荷を待たずに対応を行う。出荷依頼では出荷するリンゴの個数が指定されており、JOI 農場は出荷依頼が来た時、出荷依頼で指定されている個数のリンゴを、JOI 農場の決まりを満たす組み合わせで出荷できる場合、すぐにそれらのリンゴを出荷する。一方、出荷できない場合は、出荷できないことを伝えリンゴは全く出荷しない。

今年は JOI 農場に多くのリンゴの入荷と多くの出荷依頼が来ることを見込まれている。そこで、あなたに JOI 農場のためのリンゴ管理プログラムを書いて欲しい。

課題

各々のリンゴには色の濃さが整数値で決まっている。1 回の出荷におけるリンゴの濃さのバラつきとは、出荷されるリンゴの濃さの最大値と最小値の差である。JOI 農場では各出荷依頼について、濃さのバラつきが B 以下でなければならない、という決まりがある。

JOI 農場が行うリンゴ管理プログラムへは合計 M 個の要求が行われる。 i 番目の要求は「リンゴの入荷」「出荷依頼」「プログラムの終了」の 3 種類のうちのいずれかである。それぞれの要求に対し、リンゴ管理プログラムは以下の動作をしなければならない。

- 「リンゴの入荷」の要求に対しては、濃さの値が D_i のリンゴを 1 個入荷したことが入力されるので、これを記憶する。
- 「出荷依頼」の要求に対しては、 N_i 個のリンゴを出荷する出荷依頼が来たことが入力されるので、出荷できる場合は出荷するリンゴの濃さの値をそれぞれ昇順に出力し、出荷できない場合は **NO** を出力する。ただし、出荷するリンゴの組み合わせの候補が複数存在する場合は、リンゴの色の濃さの値の総和が最も大きくなる組み合わせを選択する。
- 「プログラムの終了」の要求に対しては、リンゴ管理プログラムの終了命令が入力されるので、プログラムを終了する。

注意

この課題は応答型課題 (Reactive task) である。 i 番目の要求が「出荷依頼」である場合、 $i+1$ 番目の要求の入力は i 番目の出荷依頼に対する出力をした後に与えられる。

また、出力がバッファされ、読み込みが誤ってブロックされてしまうことを防ぐ必要がある。そのため読み込みを行う前に `fflush(stdout);` などを記述しておくことを勧める。



制限

| | |
|--|--------------------------|
| $1 \leq M \leq 100\,000$ | プログラムに渡される要求の数 |
| $0 \leq B \leq 1\,000\,000\,000$ | 各出荷毎のリンゴの濃さのバラつきの上限值 |
| $0 \leq D_i \leq 1\,000\,000\,000$ ($1 \leq i \leq M$) | i 番目の要求で入荷するリンゴの色の濃さの値 |
| $1 \leq N_i \leq 100\,000$ ($1 \leq i \leq M$) | i 番目の要求で出荷するリンゴの個数 |

入力

標準入力から以下の入力を読み込め.

- 1 行目には整数 M, B が空白を区切りとして書かれており, 要求の個数が M 個, JOI 農場の決まりである, リンゴの濃さのバラつきの上限值が B であることを表す.
- $1+i$ 行目 ($1 \leq i \leq M-1$) には「リンゴの入荷」「出荷依頼」のどちらかの要求を示す情報が書かれている.
 - i 番目の要求が「リンゴの入荷」である場合, 文字 **A** と, 入荷したリンゴの色の濃さの値を表す整数 D_i が空白を区切りとして書かれている.
 - i 番目の要求が「出荷依頼」である場合, 文字 **R** と, 出荷依頼の依頼内容である出荷するリンゴの個数を表す整数 N_i が空白を区切りとして書かれている.
- $1+M$ 行目には「プログラムの終了」の要求を表す文字 **E** のみが書かれている.

出力

各「出荷依頼」の要求を受けた時に, 出荷できない場合は文字列 **NO** を 1 行で出力せよ. 出荷できる場合は, 出荷するリンゴの色の濃さを昇順で空白を区切りとして 1 行に出力せよ.

採点基準

採点用データのうち, 配点の 30%分については, $B \leq 10\,000$ かつ $D_i \leq 10\,000$ ($1 \leq i \leq M$) を満たす.



入出力の例

| 入力例 | 出力例 |
|-----------------------------|-----------|
| 22 10 A 5 A 16 R 2 | |
| ----- | NO |
| A 10 R 2 | |
| ----- | 10 16 |
| R 2 | |
| ----- | NO |
| A 15 A 5 R 2 | |
| ----- | 5 15 |
| A 5 R 2 | |
| ----- | 5 5 |
| A 0 A 10 R 1 | |
| ----- | 10 |
| A 10 A 10 R 4 | |
| ----- | NO |
| A 30 R 4 | |
| ----- | NO |
| A 0 R 4 | |
| ----- | 0 0 10 10 |
| E | |



本棚 (Bookshelf)

20XX年、IOIがJOI君の住む国で開催されることになった。このニュースを聞いたJOI君は友人に知らせるために部屋を飛び出そうとした。しかし、あまりにも急いでいたため部屋の本棚にぶつかってしまい、本は本棚から全て落ちてしまった。とにかく急いでいたJOI君は、落ちた本を順番を全く考慮せず全て本棚に入れて家を出た。帰宅後、JOI君は滅茶苦茶な順番に並んだ本を本来の順番に戻す作業が必要になった。

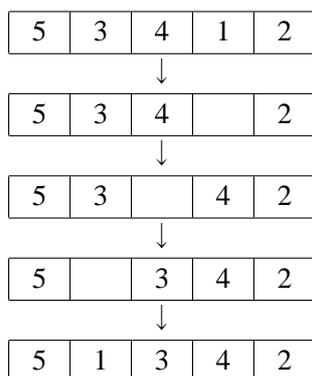
JOI君の部屋の本棚は幅 N センチメートルであり、本棚には幅1センチメートルの本が N 冊入っている。本には1から N までの番号がふられており、本来は本棚には左から $1, \dots, N$ の順番で本が並べられていた。また、本 i の重さは A_i グラムである。

JOI君は疲れていて本を同時にたくさん持ちたくなかったので、次のような操作を行うことで本棚を整理することにした:

- 本棚に入っている本を1つ選び、取り出す。
- 次に、本を取り出してできた隙間に隣接している本を動かすことを何度か行う。
- 次に、取り出した本を本棚の隙間に戻す。

本を本棚から同時に2冊以上取り出すことはできない。

例えば、本が左から5, 3, 4, 1, 2の順で並んでいるとき、本1を取り出し、本4と本3を順に右に動かし、本1を本棚に戻すことで、本棚の本を左から5, 1, 3, 4, 2の順にすることができる。(下図)



この操作で、JOI君が重さ w グラムの本を本棚から取り出すとき、JOI君はちょうど w カロリーを消費する。また、JOI君が重さ w グラムの本を本棚に戻すときも、JOI君はちょうど w カロリーを消費する。また、本棚はなめらかな素材でできているので、JOI君が本棚の中で本を動かすときにはJOI君はカロリーを消費しないとしてよい。

JOI君は疲れていたもので、なるべくカロリーを消費しないようにして本棚の本を本来の順番に戻すことにした。



課題

本の数と、それぞれの本の重さ、現在の本棚の本の並び順が与えられたとき、JOI君が本棚の本を本来の順番に並び替えるために消費するカロリーの合計の最小値を求めるプログラムを作成せよ。

制限

- $1 \leq N \leq 100\,000$ 本の数
 $1 \leq A_i \leq 1\,000\,000\,000$ 本 i の重さ (グラム)

入力

標準入力から以下の入力を読み込め。

- 1行目には整数 N が書かれている。
- 続く N 行には本の重さの情報が書かれている。 $i+1$ 行目 ($1 \leq i \leq N$) には、本 i の重さを表す整数 A_i が書かれている。
- 続く N 行には現在の本棚の本の並び順の情報が書かれている。 $j+N+1$ 行目 ($1 \leq j \leq N$) には、現在左から j 番目にある本の番号を表す整数が書かれている。

出力

標準出力に、JOI君が消費するカロリーの合計の最小値を表す整数を1行で出力せよ。

採点基準

採点用データのうち、配点の30%分については、 $N \leq 5\,000$ を満たす。

採点用データのうち、配点の30%分については、 $1 \leq i \leq N$ なる全ての i に対し $A_i = 1$ を満たす。

採点用データのうち、配点の20%分については、これら2つの条件の両方を満たす。

採点用データのうち、配点の40%分については、これら2つの条件の少なくとも一方を満たす。



入出力の例

| 入力例 | 出力例 |
|---|-----|
| 4 1 6 4 3 3 4 2 1 | 14 |

この入力例では、最初本は左から 3,4,2,1 の順に並んでおり、本 1,2,3,4 の重さはそれぞれ 1,6,4,3 グラムである。

JOI 君は、まず以下の操作を順番に行う：

- 本 1 を取り出す。
- 本 2, 本 4, 本 3 を順に右に動かす。
- 本 1 を隙間に戻す。

本棚の本は左から 1,3,4,2 の順になる。本 1 の重さは 1 グラムなので JOI 君は $1 \times 2 = 2$ カロリーを消費する。

次に、以下の操作を順番に行う：

- 本 2 を取り出す。
- 本 4, 本 3 を順に右に動かす。
- 本 2 を隙間に戻す。

本棚の本は左から 1,2,3,4 の順になる。本 2 の重さは 6 グラムなので JOI 君は $6 \times 2 = 12$ カロリーを消費する。

よって JOI 君は合計 14 カロリーの消費で本を左から 1,2,3,4 の順にできる。また、JOI 君が消費するカロリーをこれより少なくすることは不可能である。



国際情報オリンピック (IOI)

20XX年、ついにJOI国で行われることになったIOIには K 人の選手が参加した。選手には $1, 2, \dots, K$ と番号が付けられている。問題は全部で N 問出題され、各選手は各問題について0以上100以下の整数の点数を付けられる。

選手には N 問の合計点に応じてメダルが与えられる。メダルの与えられる詳しい条件は、たとえば金メダルについては次のように決まっている：

G を、 N 問の合計点が G 点以上の選手の人数が全体の $\frac{1}{12}$ 以上となるような最大の値とする。このとき、金メダルが与えられる条件は、 N 問の合計点が G 点以上であることである。

すでに競技が終了している問題が M 問あり、点数が確定している。IOIのウェブサイトで各選手の現在までの合計点を見ていたあなたは、金メダルを与えられることが確実な選手や金メダルを与えられる可能性のある選手がそれぞれどれだけいるのかが知りたくなった。

課題

各選手の現在までの合計点を与えられたとき、金メダルを与えられることが確実な選手、および、金メダルを与えられる可能性のある選手をそれぞれ番号順に出力するプログラムを作成せよ。

制限

| | |
|--------------------------------|---------------------|
| $1 \leq K \leq 100\,000$ | 選手の人数 |
| $1 \leq N \leq 10\,000\,000$ | 全体の問題数 |
| $0 \leq M \leq N$ | すでに終了した問題数 |
| $0 \leq P_i \leq 100 \times M$ | 番号 i の選手の現在までの合計点 |

入力

標準入力から以下の入力を読み込め。

- 1行目には整数 K, N, M が空白を区切りとして書かれており、選手の人数が K 人であること、全体の問題数が N 問であること、そのうちすでに終了した問題が M 問あることを表す。
- 続く K 行には各選手の得点を書かれている。 $i+1$ 行目 ($1 \leq i \leq K$) には、番号 i の選手の現在までの合計点を表す整数 P_i が書かれている。



出力

標準出力に以下の内容を出力せよ。

- 初めの a 行は、金メダルを与えられることが確実な選手の番号を、1 行に 1 つずつ、小さい順に列挙していなければならない。ただし a は金メダルを与えられることが確実な選手の人数である。
- 続く 1 行には文字列 ----- (ハイフンが 8 個) を出力せよ。
- 続く b 行は、金メダルを与えられる可能性のある選手の番号を、1 行に 1 つずつ、小さい順に列挙していなければならない。ただし b は金メダルを与えられる可能性のある選手の人数である。

採点基準

採点用データのうち、配点の 20%分については、 $K \leq 3000$ を満たす。



入出力の例

| 入力例 1 | 出力例 1 |
|--------|-------|
| 15 3 2 | 12 |
| 0 | ----- |
| 30 | 4 |
| 50 | 6 |
| 100 | 9 |
| 0 | 11 |
| 190 | 12 |
| 10 | 14 |
| 50 | |
| 100 | |
| 80 | |
| 90 | |
| 200 | |
| 50 | |
| 100 | |
| 0 | |
| 入力例 2 | 出力例 2 |
| 5 4 2 | ----- |
| 0 | 1 |
| 50 | 2 |
| 100 | 3 |
| 150 | 4 |
| 200 | 5 |



オリエンテーリング (Orienteering)

あなたの通う JOI 高校では、年に一度、オリエンテーリングが行われ、全校生徒が参加する。オリエンテーリングとは、地図と方位磁針を用い、山野に設置されたチェックポイントを巡る競技である。

JOI 高校のオリエンテーリングは、2 人一組のチームで参加するという特徴がある。各チームの 2 人は、指定されたスタート地点から一緒にスタートするが、その後は別々に行動をし、指定されたゴール地点を目指す。2 人がゴール地点に到達した際、それぞれのチェックポイントをチームのうちの少なくとも 1 人が訪れていないとならない。2 人は途中で同じ地点を訪れても構わない。また、2 人が途中で同じ道を通っても構わない。

JOI 高校のオリエンテーリングが行われる JOI 山には、 N 個の地点があり、それらの間を M 本の道が繋いでいる。 N 個の地点には 1 から N までの番号が付いている。スタートはふもとにある地点 1 であり、ゴールは頂上の地点 N である。スタートとゴール以外の地点のうち一部または全部がチェックポイントとして指定される。

JOI 高校は、混乱を避けるため、オリエンテーリングの間、それぞれの道を、高度が低い方の地点から高度が高い方の地点への一方通行とする。高度が等しい 2 つの地点は存在しない。スタートである地点 1 は最も高度の低い地点であり、ゴールである地点 N は最も高度の高い地点であるが、地点の番号は必ずしも高度の低い地点から順につけられているわけではないことに注意せよ。JOI 山では、このように道を一方通行にしても、地点 1 から全ての地点に行くことができ、また、全ての地点から地点 N に行くことができる。

課題

条件を満たすような一組のチームの 2 人の移動方法に関して、移動距離の合計の最小値を求めるプログラムを作成せよ。そのような 2 人の移動方法が存在することは保証されている。

制限

- $3 \leq N \leq 1000$ 地点の個数
- $2 \leq M \leq 10000$ 道の本数
- $1 \leq K \leq N - 2$ チェックポイントの数
- $1 \leq C_j \leq 10000$ 道 j の長さ



入力

標準入力から以下の入力を読み込め.

- 1行目には整数 N, M が空白を区切りとして書かれており, 地点の数が N 個, 道の数が M 本であることを表す.
- 続く N 行には各地点がチェックポイントであるかどうかの情報が書かれている. $1+i$ 行目 ($1 \leq i \leq N$) には, 0 か 1 の整数 S_i が書かれており, S_i が 0 の場合, 地点 i はチェックポイントではなく, S_i が 1 の場合, 地点 i はチェックポイントであることを表す. 常に $S_1 = S_N = 0$ である.
- 続く M 行には道の情報が書かれている. $1+N+j$ 行目 ($1 \leq j \leq M$) には, 整数 A_j, B_j, C_j が空白を区切りとして書かれており, 道 j は地点 A_j から地点 B_j を結び, 道 j の長さが C_j であることを表す. 地点 A_j は地点 B_j より高度が低い地点であり, 道 j は地点 A_j から地点 B_j への一方通行となるものとする. 必ず $A_j \neq B_j$ であり, また, $A_j = A_k$ かつ $B_j = B_k$ なる道 k ($k \neq j$) は存在しない.

出力

条件を満たすような一組のチームの 2 人の移動方法に関する, 移動距離の合計の最小値を出力せよ.

採点基準

以下, チェックポイントの数を K とおく. すなわち, $K = S_1 + S_2 + \dots + S_N$ である.

採点用データのうち, 配点の 30% 分については, $K \leq 10$ を満たす.

採点用データのうち, 配点の 30% 分については, $N \leq 100$ かつ $M \leq 500$ を満たす.

採点用データのうち, 配点の 10% 分については, これら 2 つの条件の両方を満たす.

採点用データのうち, 配点の 50% 分については, これら 2 つの条件の少なくとも一方を満たす.



入出力の例

| 入力例 | 出力例 |
|--|-----|
| 8 12 0 1 0 0 1 1 0 0 1 4 5 1 6 5 4 2 4 4 7 9 4 5 6 2 5 8 2 8 3 6 2 7 6 7 8 7 3 2 3 5 7 5 8 3 | 29 |

この入力例は、下図に対応する。チェックポイントとなっている地点は水色で示されている。また、最小の移動距離を達成する2人の移動方法は赤色で示されている。

