

に・ぜろ・いち・ろく

# パソコン甲子園2016

## 全国高等学校パソコンコンクール プログラミング部門 本選問題

平成28年11月12日(土) 午後1時45分～午後5時45分



全国高等学校パソコンコンクール実行委員会

## 問題 1 長方形

(3点)

アイズ放送協会の教育番組(AHK教育)では、子ども向けの工作番組「あそんでつくる」を放送しています。今日は棒で長方形を作る回ですが、用意した4本の棒を使って長方形ができるかを確認したいと思います。ただし、棒は切ったり折ったりしてはいけません。

### 課題

4本の棒の長さが与えられるので、それらすべてを辺とする長方形が作れるかどうか判定するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

```
e1 e2 e3 e4
```

入力は1行からなり、各棒の長さを表す整数 $e_i$  ( $1 \leq e_i \leq 100$ )が与えられる。

### 出力

長方形を作成できる場合には「yes」を、作成できない場合には「no」を出力する。ただし、正方形は長方形の一種なので、正方形の場合でも「yes」と出力する。

### 入出力例

入力例 1	出力例 1
1 1 3 4	no
入力例 2	出力例 2
1 1 2 2	yes
入力例 3	出力例 3
2 1 1 2	yes
入力例 4	出力例 4
4 4 4 10	no

## 問題2 棒で作る直方体

(4点)

アイズ放送協会の教育番組(AHK教育)では、子ども向けの工作番組「あそんでつくる」を放送しています。今回は棒で箱を作る回ですが、用意した12本の棒を使って直方体ができるかを確認したいと思います。ただし、棒は切ったり折ったりしてはいけません。

### 課題

12本の棒の長さが与えられるので、それらすべてを辺とする直方体ができるかどうか判定するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

```
e1 e2 ... e12
```

入力は1行からなり、各棒の長さを表す整数 $e_i$  ( $1 \leq e_i \leq 100$ )が与えられる。

### 出力

直方体を作成できる場合には「yes」を、作成できない場合には「no」を出力する。ただし、立方体は直方体の一種なので、立方体の場合でも「yes」と出力する。

### 入出力例

入力例1	出力例1
1 1 3 4 8 9 7 3 4 5 5 5	no
入力例2	出力例2
1 1 2 2 3 1 2 3 3 3 1 2	yes

### 問題3 有理式最大化

(5点)

N個の異なる自然数が与えられる。その中から異なる4つを選んで、それらをA, B, C, Dとしたとき、次の数式

$$\frac{A+B}{C-D}$$

の最大値を求めたい。上式のA, B, C, Dに、与えられた自然数のすべての組み合わせを代入して、最大値を変数maxvalに記録する方式の処理例を以下に示す。ただし、与えられた自然数はdata[0], data[1], ..., data[N-1]に格納されている。

```
double maxval=0;
for(a=0; a<N; ++a) {
  for(b=0; b<N; ++b) {
    if(b==a) continue;
    for(c=0; c<N; ++c) {
      if(c==a || c==b) continue;
      for(d=0; d<N; ++d) {
        if(d==a || d==b || d==c) continue;
        double val=(double)(data[a]+data[b])/(data[c]-data[d]);
        if(val>maxval) maxval=val;
      }
    }
  }
}
```

この処理は、個数Nが小さいときは直ちに終了するが、Nが大きくなると結果を得るまでに時間がかかるようになるので望ましくない。なぜならば、ループの最も内側の計算は、 $N \times (N-1) \times (N-2) \times (N-3)$ 回繰り返されるためである。たとえば、Nが1000になると、この処理のループの最も内側の計算は約 $10^{12}$ （1兆）回繰り返されることになり、計算の速いコンピュータでもすぐに答えを得ることはできない。ところが、上より効率の良い処理を行うプログラムを作成することにより、Nが1000程度であっても短い時間で答えを得ることができる。

#### 課題

N個の異なる自然数が与えられたとき、その中から異なる4つを選んで、上の数式の最大値を求めるプログラムを作成せよ。ただし、上の処理をそのまま利用したプログラムを提出すると、時間制限で誤答となることに注意せよ。

#### 入力

入力は以下の形式で与えられる。

```
N
a1 a2 ... aN
```

1行目に自然数の個数N( $4 \leq N \leq 1000$ )が与えられる。2行目に各自然数の値 $a_i$  ( $1 \leq a_i \leq 10^8$ )が与えられる。ただし、同じ自然数が重複して現れることはない ( $i \neq j$ について $a_i \neq a_j$ )。

#### 時間制限

入力に対して、実行時間が2秒を超えてはならない。

## 出力

与えられた  $N$  個の自然数に対して、上の数式の最大値を実数で出力する。ただし、誤差がプラスマイナス  $10^{-5}$  を超えてはならない。この条件を満たせば、小数点以下を何桁表示してもよい。

## 入出力例

入力例 1	出力例 1
10 1 2 3 4 5 6 7 8 9 10	19.00000

入力例 1 では、 $A=9$ ,  $B=10$ ,  $C=2$ ,  $D=1$  などの組み合わせで最大になる。この場合は、小数点以下を省略して「19」と出力しても良い。

入力例 2	出力例 2
5 22 100 42 3 86	9.78947

入力例 2 では、 $A=100$ ,  $B=86$ ,  $C=22$ ,  $D=3$  などの組み合わせで最大になる。

入力例 3	出力例 3
6 15 21 36 10 34 5	18.00000

入力例 3 では、 $A=21$ ,  $B=15$ ,  $C=36$ ,  $D=34$  などの組み合わせで最大になる。

入力例 4	出力例 4
4 100000 99999 8 1	28571.285714

※ C++の `iostream` の `cout` を使う場合、`cout` を行う前に以下の関数を呼び出せば、小数点以下6桁までを出力させることができる。

```
std::cout.setf(std::ios_base::fixed, std::ios_base::floatfield);
```

## 問題4 必勝7並べ

(8点)

トランプを使ったゲームに「7並べ」があります。ここではそれを簡単にしたゲームを考えます。1から13の番号がそれぞれ書かれた13枚のカードを使って7並べをします。対戦は、2者だけで次のようにゲームを進めます。

1. 「場」に7のカードを置きます。
2. 2者には、残りのカードがランダムに6枚ずつ配布されます。
3. 先手の手持ちのカードのうち、場にあるカードの番号と連続する番号のカードがあれば、そのうちの1枚を場に置きます。プレイヤーはカードが置ける場合には必ず置かなければいけません。無いときに限り、カードを出さずに相手の番になります。
4. 後手も同じ要領で、手持ちのカードを場に置きます。
5. 手順3と4を繰り返して、一方の手持ちのカードがなくなるまで続けます。先に手持ちのカードをすべて場に置けた方が勝者となります。

### 課題

先手のカードの番号が与えられたとき、後手がどのようにカードを出してきても、先手が勝つ手順が少なくとも一つあるかを判定して出力するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

```
N
game1
game2
:
gameN
```

1行目には、ゲームを行う回数 $N$  ( $1 \leq N \leq 100$ ) が与えられる。続く $N$ 行に、 $i$ 回目のゲームの情報 $game_i$  が与えられる。各 $game_i$ は、以下の形式で与えられる。

```
f1 f2 f3 f4 f5 f6
```

$f_j$  ( $1 \leq f_j \leq 13, f_j \neq 7$ ) は先手に配られるカードの番号である。ただし、同じ行に番号が重複して現れることはない ( $j \neq k$  について  $f_j \neq f_k$ )。

### 出力

各ゲームについて、後手がどのようにカードを出してきても、先手が勝つ手順が少なくとも一つある場合「yes」、そうでない場合「no」と1行に出力する。

### 入出力例

入力例	出力例
5	yes
1 2 3 4 5 6	yes
1 3 5 6 8 4	no
1 2 3 4 5 8	yes
1 2 4 5 10 11	no
1 2 3 6 9 11	



## 問題5 環状すごろく

(8点)

ユキさんは、子供会の催しでみんなで遊べるようにすごろくを作りました。このすごろくでは、環状にマスが並んでいて、それぞれのマスには1以上の整数が書いてあります。

プレイヤーは出発点としてどこかのマスを選んで自分の駒を置きます。そのマスに書いてある数だけ、時計回りに駒を進めます。止まったマスに書いてある数だけ、再び時計回りに駒を進めます。これを繰り返して、出発点に選んだマスの上で駒が止まったら「あがり」です。

実際には、マスの選び方によっては絶対に「あがり」にならない場合もあります。ユキさんは、このすごろくで「あがり」にたどり着けるマスの個数を数えようとしています。

### 課題

すごろくの情報を入力とし、「あがり」にたどり着けるマスの個数を報告するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

```
N
a1 a2 ... aN
```

1行目にすごろくに含まれるすべてのマスの個数  $N$  ( $1 \leq N \leq 100000$ ) が与えられる。2行目に、それぞれのマスに書かれた数  $a_i$  ( $1 \leq a_i \leq 10^9$ ) が、時計回りに順番に与えられる。

### 出力

「あがり」にたどり着けるマスの個数を1行に出力する。

### 入出力例

入力例 1	出力例 1
3 1 1 1	3
入力例 2	出力例 2
3 1 1 2	2
入力例 3	出力例 3
8 2 3 7 3 3 3 4 4	6



## 問題6 実数既約分数化

(8点)

実数のうち、小数部が循環するものと有限桁のものは分数として表すことができます。

### 課題

分数で表すことができる実数が与えられたとき、その実数と等しい既約分数（それ以上約分できない分数）を出力するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

str
-----

1行に、変換したい実数を表す文字列strが与えられる。実数の値は0より大きい。文字列は数字か「.」、「(」、「)」を含む、長さが3以上8以下の文字列である。「.」は小数点、「(」は数字の循環の始まり、「)」は数字の循環の終わりを示す。整数部にも小数部にも、必ず1桁以上の数字が与えられるとする。

ただし、循環小数が与えられた場合、文字列は以下の条件を満たす。

- 循環の始まりと終わりのペアは、小数点の右側に一度だけ現れる。
- 循環の終わりを示す「)」は、文字列の末尾に現れる。
- 循環の始まりと終わりの間には、必ず1桁以上の数字が与えられる。

### 出力

実数を既約分数で表した形式（分子の整数に続けて「/」区切りで分母の整数を並べたもの）で出力する。

### 入出力例

入力例 1	出力例 1
0.(3)	1/3
入力例 2	出力例 2
1.0	1/1
入力例 3	出力例 3
5.2(143)	52091/9990
入力例 4	出力例 4
0.0739	739/10000

## 問題7 静かな町

(10点)

アイズ国では、毎年、駅伝大会が開催されています。アイズ国には $N$ 個の町が点在しており、それぞれ1から $N$ までの番号が付いています。いくつかの町の間は、互いに直接行き来できる道でつながっています。また、どの町の間もいくつかの道を辿って行き来ができます。大会のコースは、次のようにして決めます。

- 2つの町のすべての組み合わせについて、最短経路の距離を求める。
- それらの中で、最短経路の距離が最大になるような2つの町を、スタートの町とゴールの町とする。町の組み合わせが複数ある場合、そのうちのどれか一つを選ぶ。
- 選ばれたスタートの町とゴールの町の間 shortest 経路を大会のコースとする。最短経路が複数ある場合、そのうちのどれか一つを選ぶ。

ヤマト国から来たお坊さんのトクイツは、アイズ国のできるだけ静かな町に新しいお寺を開きたいと考えています。そのため、駅伝大会のコースに使われる可能性がない町を知りたがっています。

### 課題

アイズ国の町を結ぶ道の情報が与えられたとき、駅伝大会のコースに使われる可能性がない町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与えられる。

```
N R
s1 t1 d1
s2 t2 d2
:
sR tR dR
```

1行目に町の数 $N$  ( $2 \leq N \leq 1500$ ) と、町の間を直接つなぐ道の数 $R$  ( $1 \leq R \leq 3000$ ) が与えられる。続く $R$ 行に2つの町を双方向に直接つなぐ道が与えられる。 $s_i$  と  $t_i$  ( $1 \leq s_i < t_i \leq N$ ) は $i$ 番目の道がつなぐ2つの町の番号を表し、 $d_i$  ( $1 \leq d_i \leq 1000$ ) はその道の長さを表す。ただし、どの2つの町についても、それらを直接つなぐ道は高々1本とする。

### 時間制限

入力に対して、実行時間が6秒を超えてはならない。

### 出力

与えられた道の情報に対して、駅伝大会のコースになることがない町をすべて出力する。1行目に駅伝大会のコースになることがない町の数 $M$ を出力する。続く $M$ 行に、そのような町の番号を昇順で出力する。

## 入出力例

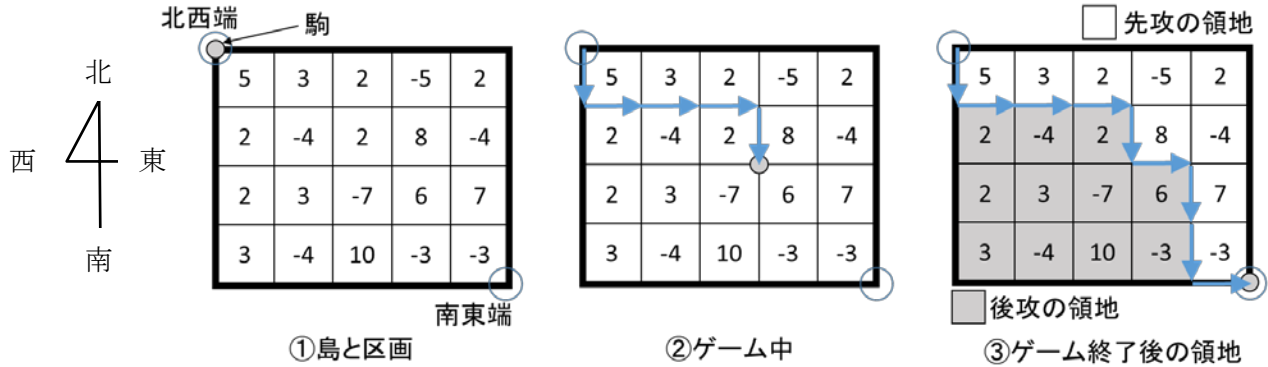
入力例 1	出力例 1
4 5 1 2 2 1 3 2 2 3 1 2 4 2 3 4 1	1 2

入力例 2	出力例 2
7 11 1 2 2 1 3 1 2 4 2 2 5 2 2 6 1 3 4 1 3 5 1 3 6 1 4 7 2 5 7 2 6 7 1	3 2 4 5

## 問題8 勢力の予想

(10点)

信夫さんと静夫さんは長方形の島で領地を取り合うゲームをしています。下の図①のように、島全体は格子状に区切られた正方形の区画できており、区画にはそこから生じる損得が整数で示されています。



このゲームでは1つの駒を動かして領地の境界線を決めていきます。ゲーム開始時、駒は島の北西端にあります(図①)。この駒を島の南東端まで動かしていったときの駒の軌跡が領地の境界線となります。二人のプレイヤーは交互に駒を動かします。駒は南隣の格子点か東隣の格子点にのみ動かすことができます(図②)。駒が島の端に到達した場合は、南か東のうち動かせる方向に動かします。駒が南東端に到達したらゲーム終了です。

ゲーム終了後の境界線から北東側の領域が先攻プレイヤーの領域、南西側の領域が後攻プレイヤーの領域です(図③)。各プレイヤーの領域内に含まれる、区画から生じる損得の合計がそのプレイヤーのスコアとなります。二人ともゲームにはかなり慣れており、最終的に自分のスコアから相手のスコアをひいた値が一番大きくなるように正確に駒を動かします。

### 課題

島の大きさとそれぞれの区画から生じる損得が与えられたとき、ゲーム終了時の結果を計算するプログラムを作成せよ。結果は、信夫さんと静夫さんのスコアの差の絶対値とする。

### 入力

入力は以下の形式で与えられる。

```

W H
S1,1 S1,2 ... S1,W
S2,1 S2,2 ... S2,W
:
SH,1 SH,2 ... SH,W
    
```

1行目に島に含まれる東西方向と南北方向の区画の数  $W, H (1 \leq W, H \leq 1000)$  が与えられる。続く  $H$  行に  $i$  行  $j$  列目の区画から生じる損得  $s_{i,j} (-1000 \leq s_{i,j} \leq 1000)$  が与えられる。ただし、 $i$  の値が増加する方向を南、 $j$  の値が増加する方向を東とする。

### 時間制限

入力に対して、実行時間が3秒を超えてはならない。

## 出力

信夫くんと静夫くんのスコアの差の絶対値を1行に出力する。

## 入出力例

入力例 1	出力例 1
2 1 -2 1	1

入力例 2	出力例 2
2 2 2 -3 3 -1	3

入力例 3	出力例 3
5 4 5 3 2 -5 2 2 -4 2 8 -4 2 3 -7 6 7 3 -4 10 -3 -3	5

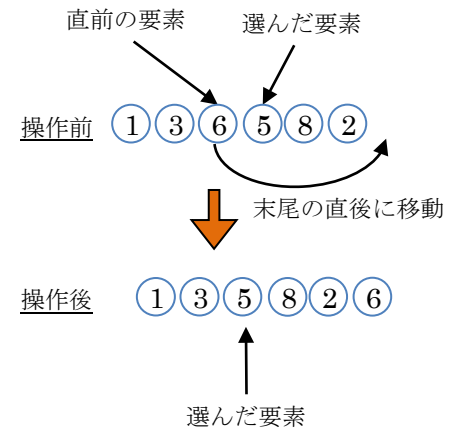
## 問題9 ソート

(11点)

高校入学後、プログラミング部に入部したタケ子さんは、次第にアルゴリズムの面白さにのめりこんでいきました。いまでは、2年生になったらプログラミング甲子園に出場してみたいと考えています。

あるとき、ソート・アルゴリズムについて学んだタケ子さんは、ソート・アルゴリズムを自分で設計してみました。タケ子さんが作ったソート・アルゴリズムでは、入力として要素の間に重複のない、1個以上の自然数からなる列が与えられたとき、以下の処理を実行します。

1. はじめに、列の先頭の要素を選ぶ。
2. 選んだ要素の直前に要素があるとき、選んだ要素とその直前の要素を比べる。直前の要素のほうが大きいなら、それを列の末尾の直後に移動させる(右図)。この操作を、選んだ要素が列の先頭になるか、選んだ要素よりその直前の要素の方が小さくなるまで続ける。
3. 選んだ要素が列の末尾なら終了。そうでなければ、選んだ要素の直後の要素を新たに選び、2へ戻る。



タケ子さんはこのアルゴリズムがどのくらいの計算時間を必要とするか見積もるために、要素を列の末尾の直後に移動させる操作の回数を数えることにしました。

### 課題

列の情報を入力とし、要素を列の末尾の直後に移動させる操作の回数を報告するプログラムを作成せよ。

### 入力

入力は以下の形式で与えられる。

```
N
a1 a2 … aN
```

1行目に列に含まれる要素の個数  $N$  ( $1 \leq N \leq 200000$ ) が与えられる。2行目に列の要素  $a_i$  ( $1 \leq a_i \leq 10^9$ ) が先頭から順番に与えられる。要素  $a_i$  に重複はない。

### 時間制限

入力に対して、実行時間が3秒を超えてはならない。

### 出力

要素を列の末尾の直後に移動させる操作の回数を1行に出力する。

## 入出力例

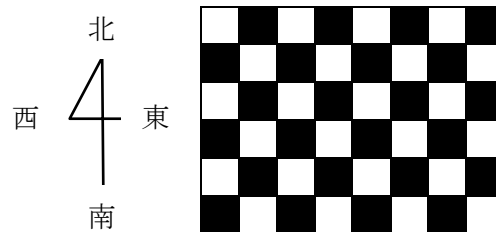
入力例 1	出力例 1
6 1 3 6 5 8 2	10

入力例 1 では、要素の移動が行われるたびに、以下のように列が変化していく。

0回目 : 1 3 6 5 8 2  
1回目 : 1 3 5 8 2 6  
2回目 : 1 3 5 2 6 8  
3回目 : 1 3 2 6 8 5  
4回目 : 1 2 6 8 5 3  
5回目 : 1 2 6 5 3 8  
6回目 : 1 2 5 3 8 6  
7回目 : 1 2 3 8 6 5  
8回目 : 1 2 3 6 5 8  
9回目 : 1 2 3 5 8 6  
10回目 : 1 2 3 5 6 8

入力例 2	出力例 2
4 4 3 2 1	6

大きなチェス盤上に、それぞれ 1 から  $N$  までの番号が割り振られた  $N$  匹の蟻がいます。図のように、チェス盤は  $H \times W$  のマスからなる長方形で、北西角を白として、白マスと黒マスが交互に並んでいます。



最初、どの蟻もチェス盤のマスの中にいて、東または南を向いています。1つのマスの中に2匹以上の蟻がいることはありません。

今、蟻たちが一斉に動き出します。すべての蟻は1単位時間に向いている方向のマスに1つ移動します。ただし、移動先がチェス盤の外の場合、落下してチェス盤から姿を消します。

チェス盤上で2匹の蟻が同じマスに入ると、それらの蟻は以下のような行動をとります：

- そのマスの色が白ならば、東方向に進んでいた蟻は南方向へ、南方向に進んでいた蟻は東方向へ向きを変える。
- そのマスの色が黒ならば、それぞれの蟻は進む方向を維持する。

### 課題

チェス盤の大きさと蟻の情報が与えられたとき、落下する順番に蟻の番号を報告するプログラムを作成せよ。ただし、同じ時刻に複数の蟻が落下する場合は、番号がより小さい方を先に報告する。

### 入力

入力は以下の形式で与えられる。

```

W H N
x1 y1 d1
x2 y2 d2
:
xN yN dN
    
```

1行目にチェス盤の東西方向のマス数  $W$  と南北方向のマス数  $H$  ( $2 \leq W, H \leq 10^9$ ) と蟻の数  $N$  ( $1 \leq N \leq 200000$ ) が与えられる。続く  $N$  行に  $i$  番目の蟻の東西方向の位置  $x_i$  ( $1 \leq x_i \leq W$ )、南北方向の位置  $y_i$  ( $1 \leq y_i \leq H$ )、向きを表す文字  $d_i$  (東向きの場合「E」、南向きの場合「S」) が与えられる。ここで、チェス盤の北西角のマスをも (1, 1)、 $x$  が増加する方向を東、 $y$  が増加する方向を南とする。

### 時間制限

入力に対して、実行時間が3秒を超えてはならない。

### 出力

落下する順番に、蟻の番号を1行ずつ出力する。



## 入出力例

入力例 1	出力例 1
3 3 3	3
2 1 S	1
1 2 E	2
2 2 E	

入力例 2	出力例 2
5 4 3	2
3 1 S	3
2 2 E	1
1 3 E	

## 問題 1 1 文字列ゲーム

(11点)

あなたは、双子のアイ君とゾー君に文字列を使ったゲームのプログラムをプレゼントしました。このゲームでは、アイ君とゾー君が文字列の中から部分文字列をそれぞれ選び、それらを比較して小さい方を選んだ人に得点が加算されます。二人は競い合い、何度もゲームを行いました。ところが、同じ文字列に対して何度もゲームをしているうちに飽きてしまいました。そこであなたは、文字列が変化するようにプログラムを修正することにしました。

### 課題

長さ  $N$  の文字列  $U$  と  $Q$  個の命令文が与えられたとき、以下の命令を処理するプログラムを作成せよ。

- 文字列  $U$  の指定された範囲にあるすべての文字を、指定された文字に置き換える。
- 文字列  $U$  の指定された 2 つの部分文字列  $S, T$  を辞書順で比較して、それらの大小関係を出力する。

### 入力

入力は以下の形式で与えられる。

```
N
U
Q
query1
query2
:
queryQ
```

1 行目に文字列の長さ  $N$  ( $1 \leq N \leq 200000$ )、2 行目に文字列  $U$  (英小文字のみを含む文字列) が与えられる。3 行目に命令の数  $Q$  ( $1 \leq Q \leq 100000$ ) が与えられる。続く  $Q$  行に  $i$  番目の命令  $query_i$  が与えられる。各  $query_i$  は、以下のいずれかの形式で与えられる。

```
set x y z
```

または

```
comp a b c d
```

`set x y z` は文字列  $U$  の  $x$  文字目から  $y$  文字目までを、指定された文字  $z$  に置き換えることを表す。ただし、 $1 \leq x \leq y \leq N$  であり、 $z$  は英小文字である。

`comp a b c d` は文字列  $U$  の  $a$  文字目から  $b$  文字目までの部分文字列を  $S$ 、文字列  $U$  の  $c$  文字目から  $d$  文字目までの部分文字列を  $T$  としたとき、文字列  $S$  と文字列  $T$  を辞書順で比較することを表す。ただし、 $1 \leq a \leq b \leq N$  かつ  $1 \leq c \leq d \leq N$  である。

### 時間制限

入力に対して、実行時間が 6 秒を超えてはならない。

### 出力

各 `comp` 命令について、 $S$  の方が小さいならば「s」、 $T$  の方が小さいならば「t」、両者が一致しているならば「e」と 1 行に出力する。

## 入出力例

入力例	出力例
13 aizualgorithm 9 comp 1 1 4 5 comp 2 6 1 5 set 9 12 b comp 9 9 10 10 comp 5 8 1 4 set 1 10 z set 11 13 x comp 8 10 1 5 comp 1 5 1 5	s t e t s e

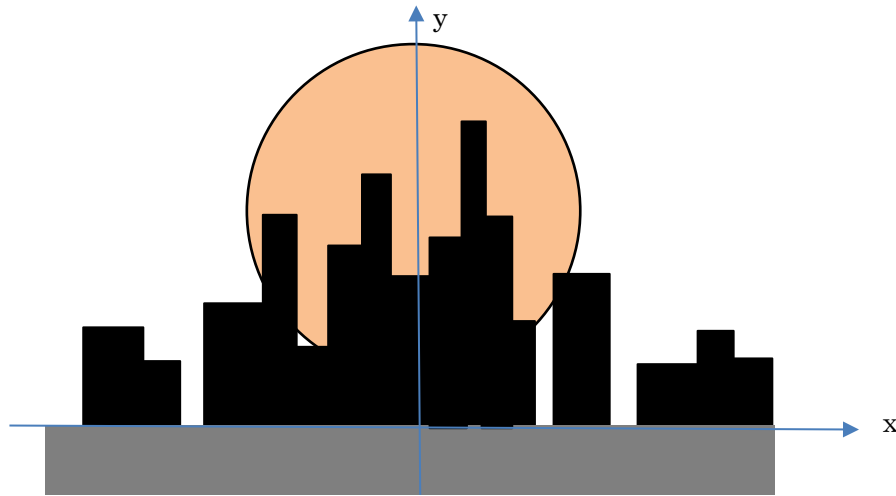
## 問題 1 2 夕暮れ

(11点)

AIZU 国の夕暮れ時は、スマートフォンを西の空へ構えて立ち止まる観光客で賑わう。AIZU 国は無数のビルが建ち並ぶ大都市であり、ビルの谷間が長く連なる西の空には、ビルのシルエットとそれらから漏れ出す太陽の光による絶景が広がる。

AIZU 観光協会の若松氏によると、太陽を表す円が、その面積のちょうど半分だけ遮られているとき、格段の絶景になるという。

図のように、西の空を、地平線を  $x$  軸、太陽の中心の軌跡を  $y$  軸とするような  $x$ - $y$  平面で表し、太陽を半径  $R$  の円、それぞれのビルのシルエットを長方形とする。



各ビルのシルエットの底辺は  $x$  軸上にあり、太陽は十分高い位置から地平線に対して垂直に沈んでいく。太陽はビルのシルエットあるいは地平線を上端とする地面に遮られ、やがて地平線の下へ消えてゆく。

### 課題

太陽の半径と各ビルのシルエットの情報が与えられたとき、太陽の面積のちょうど半分が遮られるような、最も高い太陽の高さ（中心の  $y$  座標）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与えられる。

```
N R
x1 w1 h1
x2 w2 h2
:
xN wN hN
```

1 行目にビルの数  $N$  ( $0 \leq N \leq 100$ ) と太陽を表す円の半径  $R$  ( $1 \leq R \leq 100$ ) が与えられる。続く  $N$  行に  $i$  番目のビルのシルエットの左下角の  $x$  座標  $x_i$  ( $-100 \leq x_i \leq 100$ ,  $x_i < x_{i+1}$ )、幅  $w_i$  ( $1 \leq w_i \leq 100$ )、高さ  $h_i$  ( $1 \leq h_i \leq 100$ ) が与えられる。入力はすべて整数で与えられる。

入力には以下の条件を満たす。

- ビルのシルエットが重なることはない ( $x_i + w_i \leq x_{i+1}$ )。
- 2つの隣接するビルのシルエットの高さの差は、 $R$  を超えない。
- ビルの左側 ( $x$  軸の負の方向) または右側 ( $x$  軸の正の方向) に隣接するビルがない場合、その高さは  $R$  を超えない。

## 時間制限

入力に対して、実行時間が 2 秒を超えてはならない。

## 出力

太陽の高さ (円の中心の  $y$  座標) を 1 行に出力する。ただし、誤差がプラスマイナス  $10^{-6}$  を超えてはならない。この条件を満たせば小数点以下何桁表示してもよい。

## 入出力例

入力例 1	出力例 1
0 2	0.00000000

入力例 2	出力例 2
3 2 -2 1 1 -1 1 2 0 2 1	1.25065774