



曲芸飛行 (Aerobatics)

出力のみの課題 (Output Only Task)

ピ太郎は曲芸飛行の競技に出場する予定である。この競技は、チェックポイントを通りながら飛行機を一定の高さで飛行させるものである。ここで、飛行機が飛ぶ領域は座標平面で表される。チェックポイントは N 個あり、1 から N までの番号が付けられている。チェックポイント i ($1 \leq i \leq N$) の座標は (X_i, Y_i) である。

競技中、ピ太郎は飛行機で N 個のチェックポイントを一度ずつ通らなければならない。具体的には、以下の行動を順に行う必要がある。

1. まず、 N 個のチェックポイントのうち 1 つをスタート地点として選び、ここから飛行を開始する。
2. 次の行動を $N - 1$ 回繰り返す。
まだ到達したことがないチェックポイントを 1 つ選び、現在いる地点から選んだチェックポイントに向けて直進する。
3. 最後にたどり着いたチェックポイントで、飛行を終了する。

ただし 2. では、スタート地点は既に到達したチェックポイントとしてみなすものとする。また、あるチェックポイントから次のチェックポイントに向けて飛行するとき、曲線を描いたり途中で曲がったりしてはならず、直進しなければならない。

すると、飛行機のルートは折れ線で表され、競技中に飛行機の方角を変えるタイミングが最大 $N - 2$ 回あることになる。チェックポイントにおける折れ線の角度が小さいと、高速に動く飛行機を方向転換させる角度が大きくなり、操縦に失敗するリスクが高まってしまう。

そこで、ピ太郎は、飛行機の操縦に成功する確率を上げるために、飛行開始地点と飛行終了地点を除く $N - 2$ 個のチェックポイントにおける折れ線の角度の最小値をできるだけ大きくしたい。

チェックポイントの座標が与えられたとき、折れ線の角度の最小値ができるだけ大きくなるようにチェックポイントを通る順番を決めよ。

入力

入力は以下の形式で与えられる。入力される値はすべて整数である。ただし、 Z_0 は採点のために使われる整数である。

```
N Z0
X1 Y1
⋮
XN YN
```



出力

N 行で出力せよ。 k 行目 ($1 \leq k \leq N$) には、 k 番目に到達するチェックポイントの番号 P_k ($1 \leq P_k \leq N$) を出力せよ。ただし、スタート地点は 1 番目に到達するチェックポイント P_1 とする。

提出方法

入力データ `input_01.txt`, `input_02.txt`, ..., `input_06.txt` に対する出力データ `output_01.txt`, `output_02.txt`, ..., `output_06.txt` のみを提出せよ。

制約

- $3 \leq N \leq 1000$.
- $\sqrt{X_i^2 + Y_i^2} \leq 10\,000\,000$ ($1 \leq i \leq N$).
- $(X_i, Y_i) \neq (X_j, Y_j)$ ($1 \leq i < j \leq N$).
- $1 \leq Z_0 \leq 179$.

配布ライブラリ

この課題では、3 個の点がなす角を計算する関数をライブラリとして利用できる。これはアーカイブの中に含まれるファイル `aerobatics.h` に含まれており、次のような仕様となっている。

- `double GetAngle(int xa, int ya, int xb, int yb, int xc, int yc)`
この関数は、以下のように点 A, B, C を定めたときに、角 BAC の大きさ (度) を 0 以上 180 以下の実数値で、十分小さい誤差で返すものとなっている。引数の順序を間違えないように注意すること。
 - 点 A について、引数 `xa` は点 A の x 座標、引数 `ya` は点 A の y 座標を表す。
 - 点 B について、引数 `xb` は点 B の x 座標、引数 `yb` は点 B の y 座標を表す。
 - 点 C について、引数 `xc` は点 C の x 座標、引数 `yc` は点 C の y 座標を表す。
 - ただし、点 A と点 B が同じ位置、または点 A と点 C が同じ位置にあるように引数が指定された場合の挙動は定義されない。

この課題の解答を求めるプログラムには、配布ライブラリの関数 `GetAngle` を使用しても、使用しなくてもよい。また、使用する場合は、関数 `GetAngle` の中身を改変して使用しても構わない。

なお、関数 `GetAngle` は、この課題の採点プログラムで用いられるものと同一である。



採点基準

各入力データに対し、得点は以下のように計算される。

あなたの出力が誤っている場合、すなわち出力された P_1, P_2, \dots, P_N が $1, 2, \dots, N$ の並び替えではない場合や、出力が指定された形式に沿っていない場合は、あなたの得点は 0 点となる。

あなたの出力が正しい場合、あなたの指定した飛行機のルートを折れ線で表したときの、飛行開始地点と飛行終了地点を除く $N - 2$ 個のチェックポイントにおける折れ線の角度の最小値を Z (度) とし、その入力データに対する配点を S とすると、あなたの得点は以下のように計算される。

- $Z \geq Z_0$ の場合、 S 点。
- $Z < Z_0$ の場合、 $S \times \frac{f(Z/180)}{f(Z_0/180)}$ 点。

ただし、関数 $f(\alpha)$ ($0 \leq \alpha \leq 1$) は次のように定義される。

$$f(\alpha) = 4\alpha^4 + \alpha$$

6 個の入力データの得点の合計を、小数点以下を四捨五入することで整数に丸めたものが、この課題の得点である。

各入力データにおける、 N, Z_0 の値と配点は、以下の通りである。

小課題	入力データ	N	Z_0	配点
1	input_01.txt	15	100	10
2	input_02.txt	200	143	15
3	input_03.txt	200	134	15
4	input_04.txt	1000	156	20
5	input_05.txt	1000	150	20
6	input_06.txt	1000	153	20



入出力例

入力例 1	出力例 1
7 90	5
3 1	3
2 5	1
0 2	7
-1 6	6
-3 1	4
-1 -4	2
4 -2	

チェックポイント 5, 3, 1, 7, 6, 4, 2 の順に進む場合、ビ太郎の飛行機のルートは下図のようになる。このとき、最も角が小さくなるのはチェックポイント 6 で、この角度は $68.19859\dots$ 度である。 $Z_0 = 90$ (度) なので、この通りに出力すると配点の約 61.5 % の点数が得られる。

