

W Bajtockim Parku Linowym rośnie N drzew, jedno za drugim w długim rzędzie za wejściem. Drzewo numer i rośnie w odległości x_i metrów od wejścia, a na wysokości y_i metrów znajduje się stacja, którą można wykorzystać jako początek lub koniec zjazdu linowego. Zrządzeniem losu, wszystkie liczby x_i oraz y_i są całkowite.

Dyrekcja Parku chciałaby otworzyć nowy zjazd, który byłby najbezpieczniejszy, czyli którego nachylenie jest możliwie najmniejsze. Jak wiemy z geometrii, nachylenie liny rozpiętej między stacją i a stacją j będzie ilorazem różnicy wysokości i różnicy odległości drzew, czyli będzie równe $\frac{|y_i - y_j|}{|x_i - x_j|}$ (oczywiście nie musi być liczbą całkowitą).

Wybierz zatem takie stacje, aby zminimalizować ten iloraz – napisz program, który wczyta opis pozycji możliwych stacji, wyznaczy najbezpieczniejszy zjazd linowy i wypisze numery stacji na standardowe wyjście. Nie interesuje nas jednak sytuacja, w której nachylenie byłoby równe 0, czyli między równymi wysokościami – po takim zjeździe nie dałoby się poruszać.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N ($1 \leq N \leq 200\,000$) określająca liczbę drzew. W kolejnych N wierszach znajduje się opis pozycji kolejnych drzew. Opis pozycji drzewa numer i składa się z dwóch liczb nieujemnych całkowitych x_i oraz y_i ($1 \leq x_i, y_i \leq 200\,000$) określających odpowiednio: odległość drzewa od wejścia w metrach oraz wysokość w metrach na jakiej została wyznaczona możliwa stacja zjazdu. Drzewa **nie muszą** być podane w kolejności rosnących x_i – były sadzone w różnych momentach, przez co otrzymały numery w przypadkowej kolejności.

Wartości x_i są parami różne (żadne dwa drzewa nie są w tym samym miejscu). Gwarantowane jest, że istnieje przynajmniej jedna para drzew o różnych wysokościach.

Wyjście

W pierwszym (jedynym) wierszu wyjścia należy dwie liczby całkowite i, j , ($1 \leq i, j \leq N$, $i \neq j$) oznaczające numery drzew, między którymi należy rozpiąć najbezpieczniejszy zjazd linowy. Jeśli istnieje wiele poprawnych odpowiedzi możesz wypisać dowolną z nich. Możesz też wypisać liczby i, j w dowolnej kolejności.

Ocenianie

Możesz rozwiązać zadanie w kilku prostszych wariantach – niektóre grupy testów spełniają pewne dodatkowe ograniczenia. Poniższa tabela pokazuje, ile punktów otrzyma Twój program, jeśli przejdzie testy z takim ograniczeniem.

Dodatkowe ograniczenia	Liczba punktów
$N \leq 1\,000$	35
dla każdego drzewa: $y_i \leq 1\,000$	50

Przykład

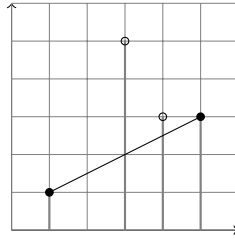
Wejście dla testu tyr0a:

```
4
1 1
4 3
3 5
5 3
```

Wyjście dla testu tyr0a:

```
4 1
```

Wyjaśnienie do przykładu: Sytuację obrazuje poniższy rysunek:



Wejście dla testu tyr0b:

```
4
1 3
4 3
6 2
9 1
```

Wyjście dla testu tyr0b:

```
1 3
```

Pozostałe testy przykładowe

- test tyr0c: $N = 200\,000$, i -ty punkt ($1 \leq i \leq N$) ma współrzędne (i, i)
- test tyr0d: $N = 200\,000$, i -ty punkt ($1 \leq i \leq N$) ma współrzędne $(i, (i^2 + 7 \bmod N) + 1)$

