

E. Odcinki

Dostępna pamięć: 128 MB

Dla zadanego zbioru odcinków należy wypisać wszystkie punkty ich przecięć. Punkt przecięcia definiujemy jako punkt wspólny dwóch odcinków; żadna para odcinków ze zbioru nie będzie miała więcej niż jednego punktu wspólnego. Odcinki tworzą z osią OX (i OY również) kąt, który jest wielokrotnością 45 stopni.

Koniec odcinka należy też do odcinka, zatem za punkt przecięcia uznajemy również koniec odcinka zawierający się w innym odcinku czy też koniec odcinka równy końcowi innego odcinka.

Specyfikacja danych wejściowych

W pierwszym wierszu danych wejściowych znajduje się liczba całkowita $n \in [2, 200\,000]$. W każdym z kolejnych n wierszy znajduje się opis jednego odcinka, będący czterema liczbami całkowitymi $x_1^i, y_1^i, x_2^i, y_2^i$, gdzie (x_1^i, y_1^i) oznacza współrzędne jednego końca odcinka i , zaś (x_2^i, y_2^i) drugiego. Te cztery liczby są oddzielone pojedynczymi spacjami i każda z nich należy do przedziału $[0, 10^6]$. Wiadomo, że wszystkich punktów przecięć będzie co najwyżej 200 000.

Specyfikacja danych wyjściowych

Jeśli żadna para odcinków nie przecina się, w pierwszym i jedynym wierszu wyjścia Twój program powinien wypisać słowo BRAK.

W przeciwnym przypadku Twój program powinien wypisać współrzędne wszystkich punktów przecięć, po jednym punkcie (tj. parę współrzędnych) w każdym wierszu wyjścia. Współrzędne punktu powinny być oddzielone pojedynczą spacją. Łatwo zauważyć, że wszystkie współrzędne będą wielokrotnościami $1/2$. Wszystkie współrzędne należy wypisać z jedną cyfrą po kropce dziesiętnej, czyli każda wypisywana liczba powinna kończyć się ciągiem .5 lub .0.

Każdy punkt powinien zostać wypisany dokładnie raz. Punkty przecięć powinny być wypisane w porządku leksykograficznym, tj. punkt (x, y) powinien zostać wypisany przed (x', y') , jeśli zachodzi $x < x'$ lub zachodzi $x = x'$ i $y < y'$.

Przykład A

Wejście:

```
2
1 1 2 2
2 1 1 2
```

Wyjście:

```
1.5 1.5
```

Przykład B

Wejście:

```
6
0 0 3 3
3 0 0 3
0 1 3 1
0 2 3 2
1 0 1 3
2 0 2 3
```

Wyjście:

```
1.0 1.0
1.0 2.0
1.5 1.5
2.0 1.0
2.0 2.0
```

Przykład C

Wejście:

```
6
0 3 6 3
1 1 4 4
1 2 5 6
3 6 6 3
2 3 3 2
5 0 5 3
```

Wyjście:

```
2.0 3.0
2.5 2.5
3.0 3.0
4.0 5.0
5.0 3.0
6.0 3.0
```