

Bezienswaardigheden



Je wil op een dag een aantal toeristische hoogtepunten zien van een stad. Je hebt een lijstje opgesteld van de plaatsen die je wil bezoeken met hun coördinaten. Je wil een zo kort mogelijke toer opstellen die alle plaatsen aandoet. Je wil hierbij vertrekken van het meest linkse punt, steeds strikt naar rechts gaan tot het meest rechtse punt en dan steeds strikt naar links terugkeren tot het beginpunt (een bitonische toer).

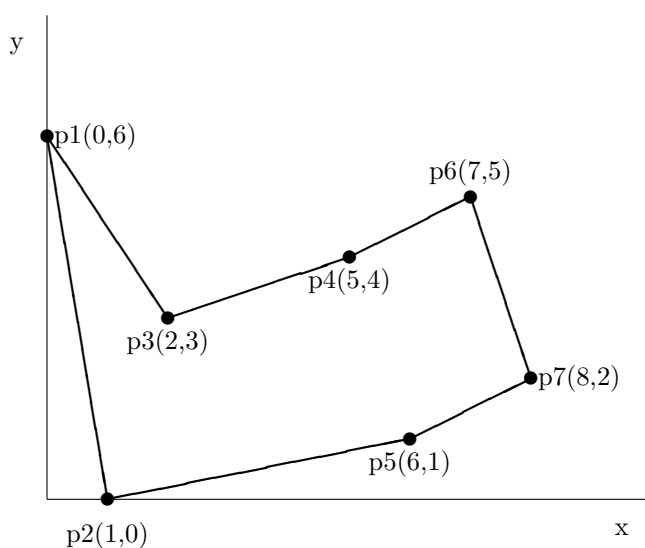
Opgave

Schrijf een programma dat voor een aantal steden (telkens met een lijst van coördinaten van hun toeristische hoogtepunten), de lengte van de kortste bitonische toer bepaalt. Neem als voorbeeld een stad met 7 bezienswaardigheden gelegen op $p_1(0,6)$, $p_2(1,0)$, $p_3(2,3)$, $p_4(5,4)$, $p_5(6,1)$, $p_6(7,5)$ en $p_7(8,2)$ zoals in onderstaande figuur. Een kortste bitonische toer gaat bijvoorbeeld als volgt:

$p_1 - p_3 - p_4 - p_6 - p_7 - p_5 - p_2 - p_1$

We nemen aan dat geen twee bezienswaardigheden dezelfde x-coördinaat hebben. Dan vormt een bitonische toer door n punten een n -hoek zodat elke verticale regel erdoorheen steeds 2 zijden doorsnijdt. Merk op dat de kortste bitonische toer niet noodzakelijk de kortste toer is. Voor het voorbeeld is $p_1-p_4-p_6-p_7-p_5-p_2-p_3-p_1$ een kortste toer die kleiner is dan de kortste bitonische toer.

De afstand tussen 2 punten is de gewone euclidische afstand.



Invoer

Alle getallen in de invoer die op dezelfde regel voorkomen, worden gescheiden door 1 enkele spatie; alle regels worden beëindigd met een enkele newline `\n`.

Op de eerste regel van de invoer staat het aantal testgevallen ($1 \leq \text{aantalSteden} \leq 100$) waarvoor de bitonische toer moet worden berekend. Voor elke stad (= voor elk testgeval) volgen dan $n+1$ regels waarvan de eerste telkens het aantal bezienswaardigheden ($2 \leq n \leq 100$) bevat. De overige n regels voor deze stad bevatten telkens de x- en y-coördinaat van de bezienswaardigheid. Deze zijn gehele getallen ($0 \leq x \leq 1000$) en ($0 \leq y \leq 1000$) en er zijn in een stad geen 2 bezienswaardigheden met dezelfde x-coördinaat. De bezienswaardigheden worden in een willekeurige volgorde gegeven.

Uitvoer

Voor elke stad is er 1 regel uitvoer: het volgnummer van het testgeval gevolgd door de afgeronde waarde van de lengte van de kortste (euclidische) bitonische toer door die stad.

Let op! Zorg ervoor dat je uitvoer geen overbodige tekens bevat, bijvoorbeeld een spatie op het einde van een regel of een lege regel op het einde van de uitvoer. Dat zorgt er immers voor dat je uitvoer als foutief wordt beschouwd.

Voorbeeld

Invoer

```
1
7
6 1
1 0
0 6
5 4
2 3
7 5
8 2
```

Uitvoer

```
1 26
```