ECE30017, Fall 2020
Problem Solving through Computational Thinking

Week 12

- **C9. Interview**
  Deadline: 11:59 PM, 19 November (Fri)
- **P9. Tumor**
  Deadline: 11:59 PM, 23 November (Tue)

C9. Interview

A company is hiring new software developers. For N applicants, document reviews and programming tests were already conducted, and the score of an applicant P_i is determined as a pair of two positive integers (x_i, y_i) where x_i is the document review score and y_i is the programming test score.

To arrange their job interviews, you are asked to group these N applicants according to the following rules:

1. Two applicants P_i and P_j must belong to a same group if (1) $x_i < x_j$ and $y_i > y_j$, or (2) $x_i > x_j$ and $y_i < y_j$, and
2. Groups must be made as many as possible

Write a program that determines the maximum number of possible applicant groups for given document review and programming test scores

Input

- Input is given as text via the standard input
- The first line has one integers N for $1 \leq N \leq 8000$.
- From the second to the $(N+1)$-th lines, each line has two integers x_i and y_i for $0 \leq x_i \leq 1000000$ and $0 \leq y_i \leq 1000000$.

Output

- Print the maximum number of possible interview groups. Your program should return the answer within 0.5 second.

Test case example

Input

```
4
8 6
15 6
20 10
14 12
```

Output

```
2
```

C9 Teams

901	이인석	권혁찬
902	정성목	차경민
903	박건희	홍순규
904	이혜림	전영우
905	박은찬	김영표
906	이수아	남진우
907	최시령	김해린
908	강석운	이찬효
909	강동인	

P9. Tumor

A new kind of cancer is recently discovered. To study its characteristics, a physiologist had cultivated N tumor cells on a tissue of 2-D plane in a Petri dish. Today the physiologist opened up the Petri dish, and found that tumor cells had been grown up in various sizes, and blood vessels are developed to connect some of tumor cells. The physiologist discovered that the blood vessels of these tumor cells have the following characteristics:

- a blood vessel is always constructed upon tissue, and
- a blood vessel is connecting only two tumor cells, and
- a blood vessel never crosses another blood vessel

The physiologist measured the weight of each tumor cell, and identified all pairs of tumor cells connected to each other with a blood vessel. A set of tumor cells forms a *tumor cluster* if every pair of these tumor cells is connected with a blood vessel. The weight of a tumor cluster is the sum of weights of its component tumor cells.

Write a program that finds the maximum tumor cluster weight for given tumor cell weights and tumor cell connections (i.e., blood vessels).

Input

- Input is given as text via the standard input
- The first line has two positive integers N and B for $2 \leq N \leq 450$ and $1 \leq B \leq 900$. The number tumor cells is N, and the number of the developed blood vessels is B.
- From the second to the (N+1)-th lines, each line gives a tumor weight between 100 and 10000. The integer at the (i+1)-th line is the weight of the i-th tumor cell.
- From the (N+2)-th to the (N+B+1)-th lines, each line contains a pair of tumor IDs that are connected by a blood vessel

Output

- Print the maximum weight of a tumor cluser. Your program should return the answer within 1.0 second.

Test case examples

Input 1

```
4 6
10
500
100
200
1 2
1 3
1 4
2 3
2 4
3 4
```

Output 1

```
810
```

Input 2

```
6 8
150
100
10
200
50
30
1 2
1 3
1 4
2 4
3 5
4 5
4 6
5 6
```

Output 2

```
450
```

Input data

- Input is given as text via the standard input
- The first line has two numbers N and B for $2 \leq n \leq 450$ and $1 \leq B \leq 900$. N is the number tumor cells, and B is the number of the developed blood vessels.
- From the second to the (N+1)-th lines, the (i+1)-th line has one integer between 100 and 10000, that represents the weight of the i-th tumor cell.
- From the (N+2)-th to the (N+B+1)-th lines, each line contains a pair of tumor ID's that are connected by a blood vessel

Output data

- Print the maximum weight of a tumor cluser. Your program should return the answer within 1.0 second.

Test case example

Input

```
4 6
10
500
100
200
1 2
1 3
1 4
2 3
2 4
3 4
```

Output

```
810
```