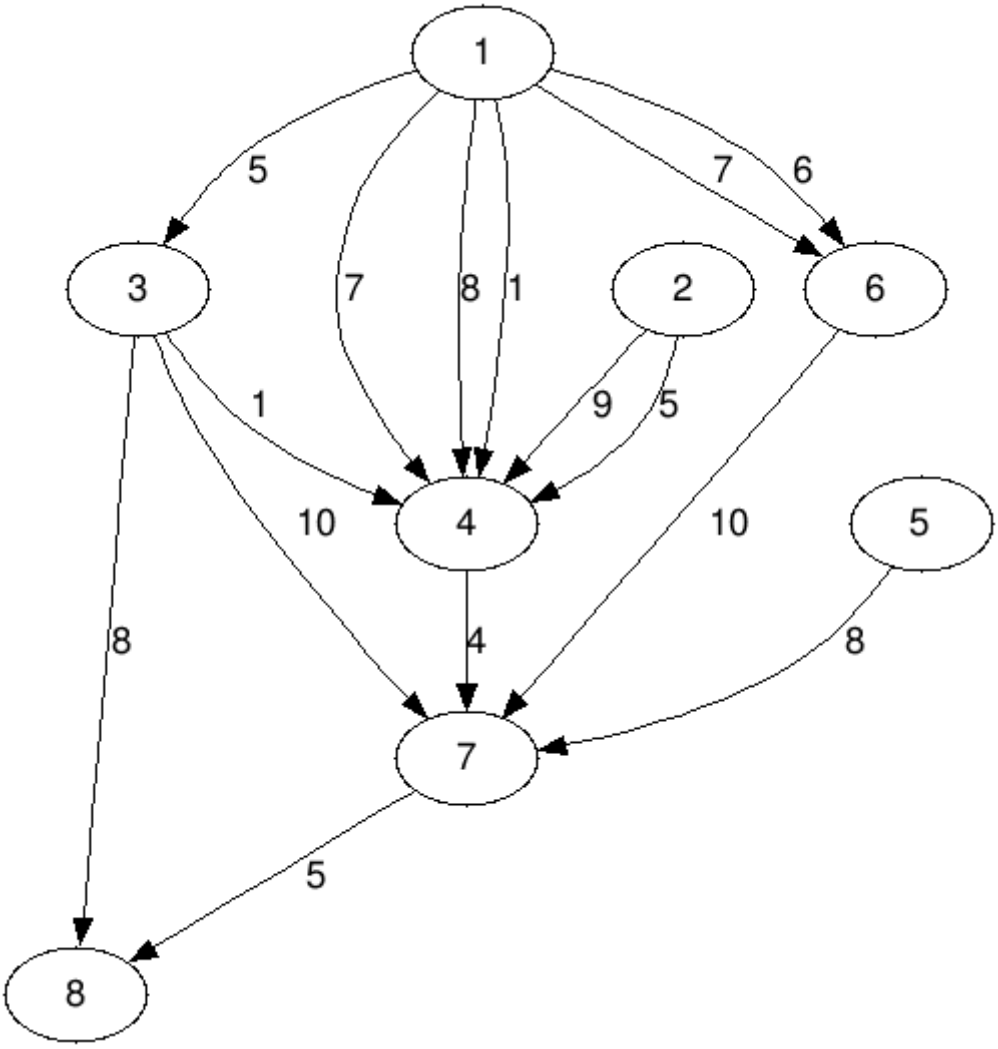
HW7

Rules & Instruction

- Compiler of programming problem.
  - C：`gcc -DONLINE_JUDGE -O2 -w -fmax-errors=3 -std=c11 main.c -lm -o main`
  - C++: `g++ -DONLINE_JUDGE -O2 -w -fmax-errors=3 -std=c++17 main.cpp -lm -o main`
  - Execution: `./main`
- Can I use theorems that haven't been mentioned in class?
  - Programming: No restriction. (無限制)
  - Handwritten: Please include its proof. (請寫上證明。)
- Can I refer to resources from the Internet or other sources that are not from textbooks or lecture slides?
  - Yes, but you must specify the references (the Internet URL you consulted with or the name and the page number of books). (可以，但必須附上參考來源，如網址或書名和頁數)
    - Handwritten: specify the references next to your solution. (手寫題請將參考附註於你的答案旁，你可以使用短網址工具)
    - Programming: specify the references at the top of your code in comments. (程式題請將參考以註解的方式附註於程式碼的最上方)
  - Although you can use external resources, it doesn't mean you can just paste the resources as your solution. **You have to answer by yourself**, and remember to specify the references; otherwise we'll view it as cheating. (雖然參考外部資源是允許的，但請用自己的話去作答，並切記要註記參考來源，否則會以作弊/抄襲處理)
- Rules of cheating.
  - Programming: We use tools to monitor code similarity. (我們用機器抓抄襲)
  - Handwritten: Though we forbid plagiarism, we still encourage you to discuss with each other. Remember that after discussion you must answer problems **in your own words**. (禁止抄襲，鼓勵大家有問題可以互相討論，**但請用自己的話去作答**。)
  - **Copycats will be scored 0.** You also need to have a cup of coffee with Professor Yeh. (**抄襲者與被抄襲者當次作業零分**，另外會跟教授喝杯咖啡☕ 。)
    - People who committed twice or more will be punished. (情節嚴重者，如累犯…，以校規處置)
- Rules of delay.
  NO LATE SUBMISSION IS ALLOWED.
  That is, zero toleration of late submission.

Non-Programming

Problem 1



(1). $(20\ points)$
Please determine the maximum flow from node 1 to node 7. Show your works, or you will get no credits.

(2). $(20\ points)$
A cut (S, T) of G is a partition of V into S and $T = V - S$.
Please determine the minimum cut of the graph above. In other words, please determine the minimum cut $(1, 7)$ of the above graph. Show your works, or you will get no credits. $(10\ points)$
Show the partition of vertices of the cut. $(10\ points)$

(3). $(15\ points)$
Briefly (within five sentences) explain why the answer to Problem 1.1 and Problem 1.2 is the same.

Problem 2

FORD-FULKERSON-METHOD(G, w, s)
1 initialize flow f to 0
2 **while** there exists an augmenting path p in the residual network G_f
3 augment flow f along p
4 **return** f

(1). $(10\ points)$
Please calculate the time complexity of pseudo code above when "finding augmenting path" is implemented by "DFS" and the graph is stored by adjacency lists.

(2). $(10\ points)$
Please calculate the time complexity of pseudo code above when "finding augmenting path" is implemented by "BFS" and the graph is stored by adjacency lists.

(3). $(25\ points)$
Follow the previous problem, please explain why causes the difference of time complexity between two flow algorithm.

Programming

pA: Network

> Time Limit: 1s
> Memory Limit: 262144 KB

Description

There are many bidirectional connections between computers in the network. As535364 (a person's name) wonders the maximum bytes can be transported from the source to the destination of the network. Moreover, there may be multiple connections between every two computers, and there is no connection between the same computer.

Input Format

The first line contains N, M, S, T represents the number of computers, the connection between computers, the ID of source, and the ID of the destination.
For the next M lines, every line contains three numbers u, v, w represent a connection between ID u_i computer and ID v_i computer with the transport restriction up to c_i bytes.

For all test data, it is guaranteed:

- $0 < N \leq 100$
- $0 < M \leq 1000$
- $1 \leq S, T, u_i, v_i \leq N \; \forall i$
- $S \neq T$
- $u_i \neq v_i \forall i$
- $0 < c_i \leq 10^3$

Subtask1 (20%)

- $0 < N \leq 15$

Subtask2 (80%)

- No other restrictions.

Output Format

Output the maximum bytes can be transported from S to T.

Sample Input

```
2 2 1 2
1 2 50
1 2 20
```

Sample Output

```
70
```

Sample Input

```
8 20 8 3
1 2 68
1 3 88
1 4 78
1 6 76
1 7 68
1 5 93
2 4 96
2 8 98
2 5 51
2 3 41
2 7 36
3 5 61
3 7 47
3 8 48
4 7 5
4 8 11
4 5 66
4 6 42
5 6 57
5 8 86
```

Sample Output

```
243
```

Hint

Tools for drawing graph

pB: Rent

> Time Limit: 1s
> Memory Limit: 262144 KB

Description

For senior students in NTNU, it isn't easy to draw the qualification for dormitory-stay. So Uier (a person's name) and his friends decide to rent apartments in New Taipei City.
Now, Uier finds two available apartments A and B, and he comes up with a way to calculate "unhappiness."

"Unhappiness" is a positive integer number that may result from the rental of the apartment or that he cannot live in the same apartment with his close friends. Please help Uier to find a way to let "unhappiness" lowest.

Note that both of the two apartments are guaranteed to accommodate all of Uier and his friends. However, it is also not necessary for them to live in the same apartment.

Input Format

The first line contains two numbers N, M represents the number of people and the number of close friend pairs.

For the second line contains N numbers $u_1, u_2, u_3 \dots u_n$ represents the value of unhappiness when $i-th$ people live in apartment A.

For the third line contains N numbers $t_1, t_2, t_3 \dots t_n$ represent the value of unhappiness when $i-th$ people live in apartment B.

For the next M lines, each of them contains three numbers a_i, b_i, c_i indicates that a_i, b_i are close friends, and if they cannot live in the same apartment, there will be c_i unhappiness.

For all test data, it is guaranteed:

- $0 < N \leq 300$
- $0 \leq M \leq C_2^n$
- $1 \leq a_i, b_i \leq N \; \forall i$
- $a_i \neq b_i \forall i$
- $0 < u_i, t_i, c_i \leq 10^4$

Subtask1 (10%)

- $M = 0$

Subtask2 (20%)

- $0 < N \leq 15$

Subtask2 (70%)

- No other restrictions.

Output Format

Please output the minimum sum of "unhappiness."

Sample Input

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

Sample Output

```
13
```

Hint

If the result is as the following table:

Person	Apartment	Unhappiness
1	A	1
2	B	$2+6$ (1, 2 aren't in the same apartment)
3	B	$1+5$ (1, 3 aren't in the same apartment)
4	A	3
5	A	$2+2$ (2, 5 aren't in the same apartment)

The sum of "unhappiness" is 22. However, the minimum sum of "unhappiness" is 13.
The following table is one of the possible partitions.

Person	Apartment	Unhappiness
1	B	4
2	B	2
3	B	1
4	A	3
5	B	3