

# Homework-4

## Maximum Flow

題目：

Network flow is a well-known difficult problem .Given a graph, your task is to find out the maximum flow and minimum cut for the weighted directed graph.

輸入：

一個檔案名：input1.txt / input2.txt/其他測資檔 (由學生手動 key)

請使用壓縮檔內的測試資料的文件檔作為你的輸入檔

檔測資檔內容：

The first line of input contains an integer T ( $1 \leq T \leq 5$ ), denoting the number of test cases.

For each test case, the first line contains two integers N and M, denoting the number of vertexes and edges in the graph. ( $2 \leq N \leq 1000$ ,  $0 \leq M \leq N^2$ ) Next M lines, each line contains three integers X, Y and C, there is an edge from X to Y and the capacity of it is C. ( $1 \leq X, Y \leq N$ ,  $1 \leq C \leq 1000$ )

輸出: (輸出在視窗中)

For each test cases, you should output the maximum flow from source 1 to sink N and the set of minimum cut edge in this case.

範例輸入 1(input1.txt)

```
3
3 2
1 2 2
2 3 1
3 3
1 2 1
```

```
2 3 1
1 3 1
6 10
1 2 16
1 3 13
2 3 10
2 4 12
3 2 4
3 5 14
4 3 9
4 6 20
5 4 7
5 6 4
```

## 範例輸出 1:

```
Case 1
max flow: 1
the source set of every cut edge:
2-3

Case 2
max flow: 2
the source set of every cut edge:
1-3
2-3

Case 3
max flow: 23
the source set of every cut edge:
2-4
5-4
5-6
```

## 範例輸出 2:

```
Case 1
max flow: 8187
the source set of every cut edge:
5 - 50
9 - 50
10 - 50
15 - 50
```

18 - 50

20 - 50

23 - 50

24 - 50

25 - 50

27 - 50

30 - 50

32 - 50

35 - 50

37 - 50

39 - 50

44 - 50

46 - 50

47 - 50

## Case 2

**max flow:** 18211

the source set of every cut edge:

1 - 5

1 - 8

1 - 9

1 - 10

1 - 15

1 - 16

1 - 17

1 - 22

1 - 27

1 - 28

1 - 29

1 - 30

1 - 33

1 - 35

1 - 38

1 - 39

1 - 40

1 - 41

1 - 51

1 - 53

1 - 54

1 - 57

1 - 58  
1 - 60  
1 - 63  
1 - 65  
1 - 67  
1 - 69  
1 - 70  
1 - 71  
1 - 75  
1 - 78  
1 - 79  
1 - 81  
1 - 84  
1 - 87  
1 - 89  
1 - 90  
1 - 91  
1 - 92  
1 - 93  
1 - 95  
1 - 97

實作兩個程式:根據 Unit.10 P.14 ,

第一題

根據 P.14 第一點請實作 the Edmonds-Karp algorithm .

✎ 若 augmenting path 搜尋方式用 BFS (稱為 the Edmonds-Karp algorithm), 則

**# iterations  $\leq nm$**  (p.729) (Time =  **$O(nm^2)$**  )

## 第二題

根據 P.14 第二點實作出另一個程式

✎ 若 augmenting path 找使 flow 增加最大之 path,  
則  $\# \text{ iterations} \leq 1 + \log_{M/(M-1)} f^*$

相當於找一最長路徑  $P$  長度為  $= \text{Min}_{e \in P} w(e)$ ,  
上式中  $f^*$  為最大 flow 的值,  $M$  為 cut 中邊數最  
多的數值 (see Sedgewick's book).

**Bonus** : 根據 P.14 第三點有寫到, 請看 Section 26.4 & 26.5.

✎ Other efficient implementations see Section 26.4  
& 26.5.

實作其演算法, 一個演算法+20, 兩個全做+40

程式語言可以使用 C 或 C++

截止時間: 6/10 (日) 晚上 23:55 前 !!!!!

作業繳交格式需繳交兩個檔案的壓縮

檔: 兩個 C 或 CPP + 報告 (PDF!!)

上傳的檔名格式: 作業名稱\_學號\_姓名.副檔名

例: HW1\_106502XXX\_王小明.c 或是 HW1\_106502XXX\_王小明.cpp

報告內容: 說明解題過程, 請比較兩種演算的差異 跟時間的快慢

如果有做 BONUS 請一起比較。

輸入輸出範例請參考說明，你可以利用本次公開測資中的資料做測試。

請在上傳前確定你的程式可以編譯且執行。

## 評分標準

編譯不過、輸出答案不對 0 分

上傳檔名不符格式 扣 5 分 (包含你的報告非 PDF 視為沒交)

輸入格式不符 扣 20 分

公開測資 1、2 都通過基本 60 分

會有一個隱藏測資(500 個點以上) 80 分

報告 加 10 分

完成一種 Bonus 加 15 分

如有抄襲 一律 0 分計算(包含網路抄襲)