

[Dashboard](#) / [My courses](#) / [Graph Theory-HK3-0405](#) / [Tuần 6 - 7 - Đường đi ngắn nhất trên đồ thị](#)
/ [Bài tập 5* - Ô kiều \(Ngưu Lang - Chức Nữ\)](#)

Started on	Tuesday, 24 June 2025, 5:51 PM
State	Finished
Completed on	Tuesday, 24 June 2025, 6:01 PM
Time taken	10 mins 2 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)



Question 1

Correct

Mark 1.00 out of 1.00

Ô kiều (Ngưu Lang - Chức Nữ)

"Ngưu Lang là vị thần chăn trâu của Ngọc Hoàng Thượng đế, vì say mê một tiên nữ phụ trách việc dệt vải tên là Chức Nữ nên bỏ bê việc chăn trâu, để trâu đi nghênh ngang vào điện Ngọc Hư. Chức Nữ cũng vì mê tiếng tiêu của Ngưu Lang nên trễ nải việc dệt vải. Ngọc Hoàng giận dữ, bắt cả hai phải ở cách xa nhau, người đầu sông Ngân, kẻ cuối sông.

Sau đó, Ngọc Hoàng thương tình nên ra ơn cho hai người mỗi năm được gặp nhau vào ngày bảy tháng bảy âm lịch. Khi tiễn biệt nhau, Ngưu Lang và Chức Nữ khóc sụt sùi. Nước mắt của họ rơi xuống trần hóa thành cơn mưa và được người dưới trần gian đặt tên là mưa ngâu." (Theo wikipedia.com)

Để gặp được nhau vào ngày 7/7, Ngưu Lang và Chức Nữ phải nhờ đến bảy quạ đen bắt cầu, gọi là Ô kiều, cho mình đi qua để gặp nhau.

Sông Ngân Hà có n ngôi sao, giả sử được đánh số từ 1 đến n . Ngưu Lang ở tại ngôi sao Ngưu (Altair), được đánh số 1, còn Chức Nữ ở tại ngôi sao Chức Nữ (Vega) được đánh số n . Để bắt được một cây cầu từ ngôi sao này sang ngôi sao kia cần một số lượng quạ nào đó. Một khi con quạ ở cây cầu nào thì phải ở đó cho đến khi Ngưu Lang và Chức Nữ gặp được nhau.

Quạ thì càng ngày càng hiếm, nên Ngưu Lang và Chức Nữ phải tính toán sao cho số lượng quạ ít nhất có thể.

Hãy giúp Ngưu Lang và Chức Nữ viết chương trình tính xem cần phải nhờ đến ít nhất bao nhiêu con quạ để bắt cầu cho họ gặp nhau.

Đầu vào (Input):

Dữ liệu đầu vào được nhập từ bàn phím với định dạng:

- Dòng đầu tiên chứa 2 số nguyên n và m , tương ứng là số ngôi sao và số cặp sao có thể bắt cầu.
- m dòng tiếp theo mỗi dòng chứa 3 số nguyên u v q nói rằng để bắt 1 cây cầu bắt qua hai ngôi sao u và v cần phải tốn q con quạ ($0 < q < 100$).

Đầu ra (Output):

- In ra màn hình số lượng quạ cần thiết.
- Xem thêm ví dụ bên dưới.

For example:

Input	Result
4 3 1 2 5 2 3 2 3 4 1	8
4 3 1 2 3 2 4 4 1 4 10	7

Answer: (penalty regime: 10, 20, ... %)

```

1 #include <stdio.h>
2 #define MAX_N 100
3 #define NO_EDGE -1
4 #define oo 999999
5
6 typedef struct
7 {
8     int n, m;
9     int W[MAX_N][MAX_N];
10 } Graph;
11
12 // ham khoi tao
13 void init_graph(Graph *pG, int n)
14 {
15     pG->n = n;
16     pG->m = 0;
17     for (int u = 1; u <= pG->n; u++)
18         for (int v = 1; v <= pG->n; v++)
19             pG->W[u][v] = NO_EDGE;
20 }
21
22 // them cung vao do thi
23 void add_edge(Graph *pG, int u, int v, int w)

```



```

23 void add_edge(Graph *pG, int u, int v, int w)
24 {
25     pG->W[u][v] = W;
26     // pG->A[v][u] = 1;
27     pG->m++;
28 }
29
30 int mark[MAX_N];
31 int pi[MAX_N]; // chiều dài từ đỉnh s đến u
32 int p[MAX_N]; // đỉnh trước của u
33
34 void MooreDijkstra(Graph *pG, int s)
35 {
36     int u, v, it;
37     for (u = 1; u <= pG->n; u++)
38     {
39         pi[u] = oo;
40         mark[u] = 0;
41     }
42
43     pi[s] = 0;
44     p[s] = -1;
45
46     // lap n-1
47     for (it = 1; it < pG->n; it++)
48     {
49         int j, min_pi = oo;
50         for (j = 1; j <= pG->n; j++)
51             if (mark[j] == 0 && pi[j] < min_pi)
52             {
53                 min_pi = pi[j];
54                 u = j;
55             }
56         mark[u] = 1; // đánh dấu đã duyệt
57
58         for (v = 1; v <= pG->n; v++)
59             if (pG->W[u][v] != NO_EDGE && mark[v] == 0)
60                 if (pi[u] + pG->W[u][v] < pi[v])
61                 {
62                     pi[v] = pi[u] + pG->W[u][v];
63                     p[v] = u; // gắn cha cho v
64                 }
65     }
66 }
67
68 int main() {
69     Graph G;
70     int n, m;
71     scanf("%d%d", &n, &m);
72     init_graph(&G, n);
73
74     for (int e = 0; e < m; e++) {
75         int u, v, w;
76         scanf("%d%d%d", &u, &v, &w);
77         add_edge(&G, u, v, w);
78     }
79
80     MooreDijkstra(&G, 1);
81     int min = 0;
82     for (int u = 1; u <= G.n; u++) {
83         if (pi[u] > min)
84             min = pi[u];
85         if (pi[u] == oo) {
86             min = -1;
87         }
88     }
89
90     printf("%d", min);
91
92
93     return 0;
94 }

```

	Input	Expected	Got	
✓	4 3 1 2 5 2 3 2 3 4 1	8	8	✓
✓	4 3 1 2 3 2 4 4 1 4 10	7	7	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[◀ Bài tập 4* - Mê cung số](#)

Jump to...



[Bài tập 7 - Thuật toán Bellman - Ford \(kiểm tra chu trình âm\) ▶](#)