**Hospital Tour, v2015 (Subtask A)**

**Released: Saturday, 19 September 2015, 11.45am**

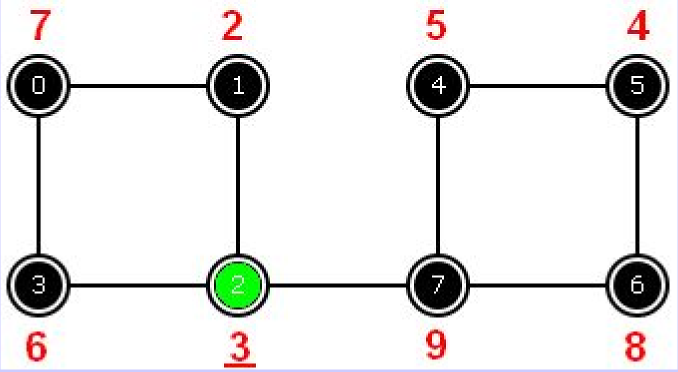
**Due: Saturday, 03 October 2015, 07.59am**

## The Actual Problem Description

This is a (fictional) story on what happened behind the scenes, many years before Steven visited that hospital…

Assume that you are the manager of your hospital. You know the layout of your hospital (a connected weighted graph) and you have given rating score to each room (the weight of each vertex of your graph). You want to decorate certain room(s) in your hospital so that visitors will feel better when visiting your hospital. As your budget is limited, you want to decorate only an *important* room with the lowest rating score. You define important room as a room that links different buildings in your hospital such that if that room (to be precise, the corridor beside that room) is blocked, the buildings in the hospital becomes 'disconnected'. You want to know the lowest rating score of an important room in your hospital.

For example, suppose your hospital is a connected weighted graph as shown below (room/vertex number is written inside the circles and the rating score of each room is written outside the circles):



A Sample Hospital Layout, try drawing this graph in VisuAlgo

There are two major buildings ({0, 1, 2, 3} and {4, 5, 6, 7}) linked by a corridor 2-7 besides the two important rooms (2 and 7). For example, if important room 2 is blocked, then people currently in room 0, 1, or 3 will not be able to visit people in rooms 4, 5, 6, or 7. Or in another word, the hospital becomes 'disconnected'. Similar situation if room 7 is blocked. The other rooms are not important rooms. For example, if room 1 is blocked, people from room 0 can still go to any other rooms in the hospital (other than room 1 of course) via path 0-3-2 and so on.

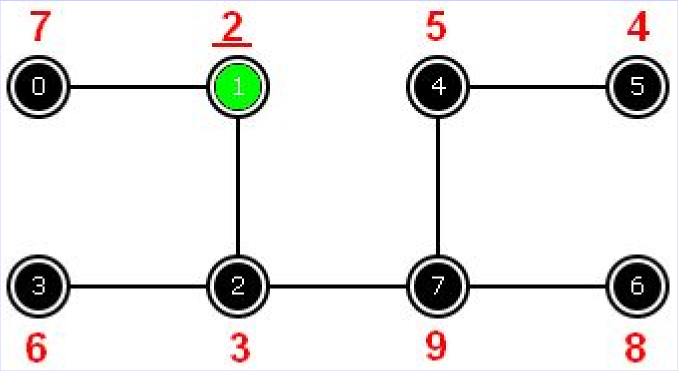
Now, among the two important rooms 2 versus 7, room 2 has lower rating score (3 points) than room 7 (9 points). So, you will decorate room 2. You will report 3 as the answer of your own query: *"What is the lowest rating score of an important room in your hospital?"*.

You just need to implement one (or more) method(s)/function(s):

- int Query()
- Query your Graph data structure and returns the lowest weight (rating score) of an important vertex (important room) of your graph (your hospital).
- These vertex weights (rating scores) are stored in another Vector of Integers.
- We guarantee that the input graph is a connected undirected unweighted graph and all rating scores are positive integers not larger than 100000.
- If your hospital has no important room, you will not decorate any room, i.e. just return -1.
- Note that the answer for this query is still unique even though there can be more than one important rooms with similar lowest rating score.

**Subtask A Constraints**

Time Limit: 1s.
The given hospital layout map is a small weighted tree (1 ≤ **V** ≤ 50) as shown in the example figure below:



A Simplified Hospital Layout (Tree), first test case in Sample Input, try drawing this tree in VisuAlgo

There are four important rooms in this hospital: 1, 2, 4, 7. Among these four rooms, important room 1 has the lowest rating score: 2. Therefore, the answer for your own query is 2.

PS: It is possible to solve Subtask A right after studying Graph Data Structures from Lecture 05 (without touching Lecture 06 content at all). Therefore, students are encouraged to implement solution for Subtask A as early as possible.

**Sample Input**

```
2

8
7 2 3 6 5 4 8 9
1 1
2 0 2
3 1 3 7
1 2
2 5 7
1 4
1 7
3 2 4 6

5
99999 1 1 1 1
4 1 2 3 4
1 0
1 0
1 0
1 0
```

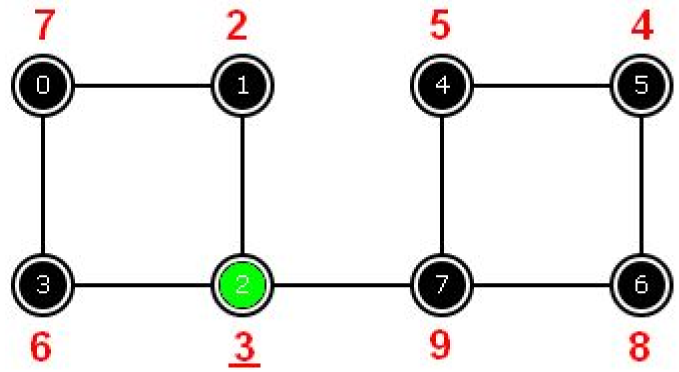
**Sample Output**

```
2
99999.
```

### Subtask B Constraints

Time Limit: 1s.

The given hospital layout map is no longer just a tree like in Subtask A, but a small, connected, and weighted general graph (1 ≤ **V** ≤ 50). Note that as a tree is also a general graph, test cases that are valid in Subtask A is also valid in Subtask B.



A Sample Hospital Layout, first test case in Sample Input, try drawing this graph in VisuAlgo

The second test case in Sample Input includes one more (bidirectional) edge: 1-4

We have more rigorous test cases to judge your final submission. If you receive Wrong Answer in this Subtask B, discuss with others about the possible corner cases. The expected solution for Subtask B requires Lecture 05 or 06 material.

**Sample Input**

```
2

8
7 2 3 6 5 4 8 9
2 1 3
2 0 2
3 1 3 7
2 0 2
2 5 7
2 4 6
2 5 7
3 2 4 6

8
7 2 3 6 5 4 8 9
2 1 3
3 0 2 4
3 1 3 7
2 0 2
3 1 5 7
```

```
2 4 6
2 5 7
3 2 4 6
```

**Sample Output**

```
3
-1
```