CS2010 2016/2017 Sem2: PS3 A

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Hospital Renovation, v2017 (Subtask A)

Released: Wednesday, 22nd February 2017, 12 noon
Due: Wednesday, 8th March 2017, 11:59 pm

You are encouraged to work with other students or teaching staffs (inside or outside this module) on solving this problem set. However, you **must** write Java code **by yourself**. In addition, when you write your Java code, you **must** list the names of every collaborator, that is, every other person that you talked to about the problem (even if you only discussed it briefly). This list may include certain posts in CS2010 Facebook group. If you have access to CS2010 files from senior batch (that is, CS2010 problem sets from year 2011-2016), please refrain from looking at their code verbatim or worse... submit your senior's code. Automatic special checks are done especially on the last/hardest subtask to compare older code with this year's version. Any deviation from this policy will be considered as cheating. If the offender is caught beyond reasonable doubt, he/she will be punished severely, including referral to the NUS Board of Discipline.

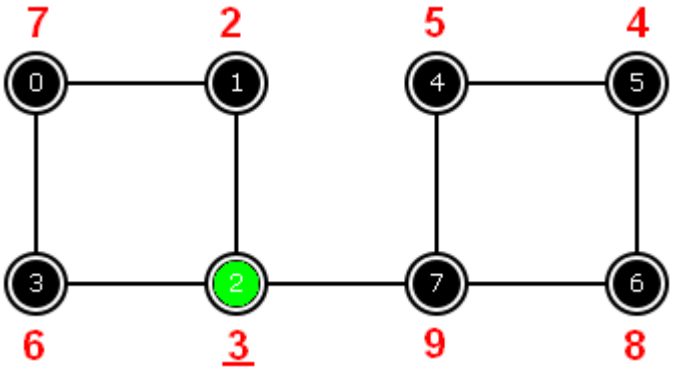
Story

Public hospitals gets government grants to upgrade their buildings after a certain number of years. In theory all the rooms would be renovated to provide the most up to date facilities and ease of accessibility for the patients. However, because the grant money is limited, only certain rooms can be renovated. One of the criterion for selection of which room to renovate is how many patients using the room over a fixed period of time (i.e the frequency of usage of the room). The more highly used a room is, the more likely it will be as a candidate for renovation. One cost-effective way to determine such rooms, is to basically to find the *critical* rooms in the hospital. A *critical* room is one that link different buildings in the hospital such that if those rooms (to be precise, the corridor beside that room) is blocked, the buildings in the hospital becomes 'disconnected'. It's reasonable to assume since such rooms connect two buildings they would receive the highest frequency of patient usage. However since it is still too costly to renovate all such *critical* rooms, all rooms in the hospital are given an integer rating, and only the *critical* room with the lowest rating score will be renovated.

The Actual Problem Description

The layout of your hospital is given as a connected weighted graph, and each room is given a rating score (the weight of each vertex of your graph). As budget is limited, only a *critical* room (as described above) with the lowest rating score can be renovated. You want to know the lowest rating score of a critical 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 critical rooms (`2` and `7`). For example, if critical 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 critical 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 critical 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 a critical room in your hospital?"*.

The skeleton program HospitalRenovation.java (click to view) that can handle all input/output details is already written for you.

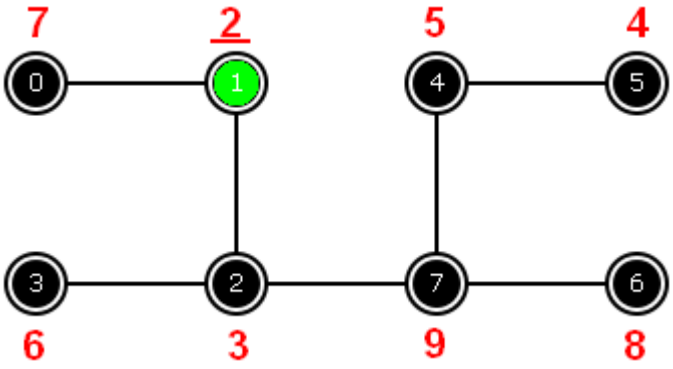
You just need to implement one (or more)[1] method(s)/function(s):

- `int Query()`
  Query your Graph data structure (already implemented in HospitalRenovation.java) and returns the lowest weight (rating score) of a critical vertex (critical room) of your graph (your hospital).
  These vertex weights (rating scores) are stored in another array 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 critical 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 critical rooms with similar lowest rating score.

Subtask A Constraints (50 points)

Time Limit: 1s.

The given hospital layout map is a small weighted tree ($1 \leq \mathbf{V} \leq 50$) 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 critical rooms in this hospital: `1`, `2`, `4`, `7`. Among these four rooms, critical 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
```

Generating Test Data

The given sample input/output are for illustration purpose and are not enough to verify the correctness of your solution.

You are encouraged to generate and post additional test data in CS2010 Facebook group.

However this time HospitalRenovationVerifier.java is not distributed as it contains partial solution to this PS. Please post your custom-made test data but only after you verify its correctness, i.e. check for these conditions before uploading them:

1. The rating scores must be positive integers not larger than 100000.
2. The input graph must be connected and undirected.
3. Make sure that the size of the graph is within the constraints stipulated in each subtask.

Problem Authors

Dr Steven Halim/Dr Chong Ket Fah
For CS2010.

Footnotes

[1] If needed, you can write additional helper methods/functions to simplify your code.

Submission (Course)

Select course: CS2010 (2016/2017 Sem 2) - Data Structures and Algorithms II ▼

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.

CS2010 2016/2017 Sem2: PS3 B

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Hospital Renovation, v2017 (Subtask B)

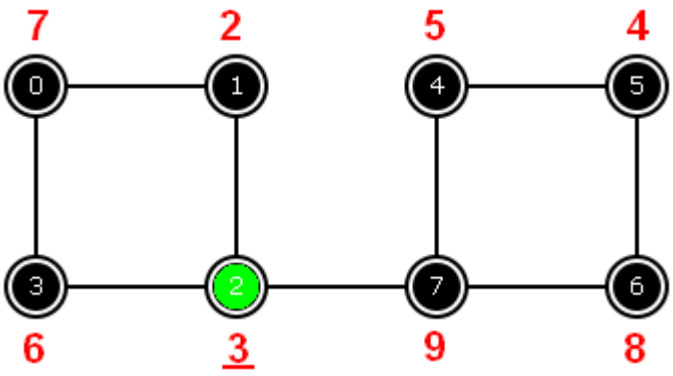
The Actual Problem Description

Please refer to Subtask A for the full problem description.

Subtask B Constraints (additional 25 points)

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 ($1 \le \mathbf{V} \le 50$). 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
```

Problem Author

Dr Steven Halim/Dr Chong Ket Fah
For CS2010.

Submission (Course)

Select course: CS2010 (2016/2017 Sem 2) - Data Structures and Algorithms II ▼

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.

CS2010 2016/2017 Sem2: PS3 C

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Hospital Renovation, v2017 (Subtask C)

The Actual Problem Description

Please refer to Subtask A for the full problem description.

Subtask C Constraints (additional 25 points)

Time Limit: 2s.

Everything is the same as with Subtask B, but the graph is larger ($1 \le \mathbf{V} \le 1000$).

Note that:

1. We have even more rigorous test cases than Subtask B to judge your final submission.
2. For this Subtask C, efficiency matters. The expected solution is $\mathbf{O(V^2 + VE)}$ per test case.
3. Our test data has $\mathbf{E = O(V)}$, i.e. $\mathbf{E = c*V}$ for a small constant factor **c**.
4. It is quite likely that your first submission is either "Wrong Answer" or "Time Limit Exceeded" for this Subtask C due to the increasing level of strictness. Therefore, do not wait until the last day to attempt Subtask C.
5. If you are unable to pass this Subtask B, you will not be able to pass Subtask C+D either as both use Subtask B test data and much more.
6. Automatic plagiarism detection is activated for this Subtask C.

Problem Author

Dr Steven Halim/Dr Chong Ket Fah
For CS2010.

Submission (Course)

Select course: CS2010 (2016/2017 Sem 2) - Data Structures and Algorithms II

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.

CodeCrunch

CS2010 2016/2017 Sem2: PS3 D

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Hospital Renovation, v2017 (Subtask D)

The Actual Problem Description

Please refer to Subtask A for the full problem description.

Subtask D Constraints (NO ADDITIONAL POINT)

Time Limit: 1s.

Your program must be able to solve Subtask C and also must be *extremely efficient* as the graph is very large (1 ≤ **V+E** ≤ 100000).

Hint: This requires a special algorithm beyond CS2010 and previously used for CS2010R. Beware that this Subtask D (which carries 0 point) can take hours to complete if you choose to attempt it.

Note that:

1. A code that solves this Subtask D should be able to solve Subtask A, B, and C easily.
2. If you are unable to pass the previous Subtask C, do not bother submitting to this Subtask D.
3. The expected solution is **O(V+E)** per test case.
4. If you encounter stack overflow, try running your Java program with: '-Xss8m' flag.

Problem Author

Dr Steven Halim/Dr Chong Ket Fah
For CS2010/R.

Submission (Course)

Select course: CS2010 (2016/2017 Sem 2) - Data Structures and Algorithms II

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.