



CodeCrunch

[Home](#) | [My Courses](#) | [Browse Tutorials](#) | [Browse Tasks](#) | [Search](#) | [My Submissions](#) | [Logout](#) | Logged in as: **e0009011**

CS2010 2016/2017 Sem2: PS5 A

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

A bleeding episode, v2017 (Subtask A)

Released: Friday, 17th of March 2017, 12:00 noon

Due: Friday, 31st of 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

Being a hemophilic, your lecturer Ket Fah will sometimes (hopefully not often) suffer from episodes of spontaneous bleeding (bleeding without trauma). This usually occur in the joint areas and the bleeding must be quickly arrested or it will become a big problem, hurt a lot and take a long time to recover. So when a bleeding episode happens, he will usually need to rush to the hospital as fast as possible to have an injection that will quickly stop the bleeding.

The Actual Problem Description

Given a map of Singapore (as a directed weighted graph), estimated time¹ to travel through Singapore roads (as *non-negative* integer weights of the corresponding directed edges -- in minutes, with value not more than 1000), Ket Fah's current position **s**, the chosen hospital **t**, determine the *shortest path* to go from Ket Fah's current position **s** to the hospital **t** that uses *no more* than **k** junctions and report the shortest path weight: The sum of edge weights along the shortest path. Ket Fah will call for a taxi or his brother will drive him and both will take this shortest path. It is guaranteed that **s** and **t** are two vertices in the given directed weighted graph. There will be **Q** queries with varying **s**, **t**, and **k** on the same given graph. If somehow there is no path from vertex **s** to vertex **t** that uses no more than **k** junctions in the given graph, output -1.

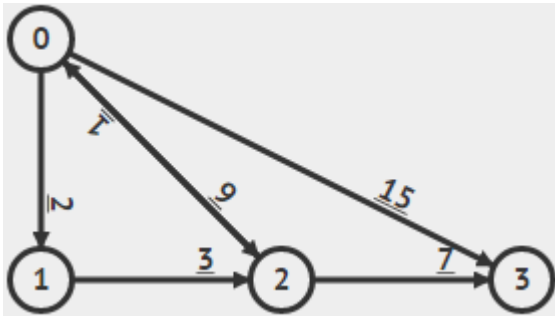
Ket Fah needs the shortest path/quickest way that is also the safest² (no more than **k** junctions).

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

You just need to implement one (or more⁴) method(s)/function(s):

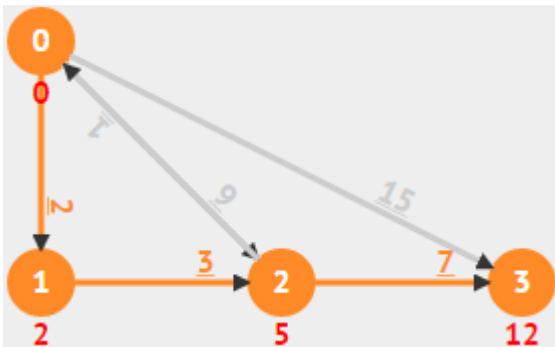
- `int Query(int s, int t, int k)`
Query the Adjacency List data structure that is already implemented in the skeleton program where the weight (in minutes) of each road (edge) is stored in the Adjacency List itself, and return the required answer.

Now, let's go back to the problem. For example, suppose Singapore³ a directed weighted graph as shown below (all edges are directed; notice that the pair of directed edges $0 \rightarrow 2$ (weight 9) and $2 \rightarrow 0$ (weight 1) that connects vertex 0 and vertex 2 can have different edge weights):



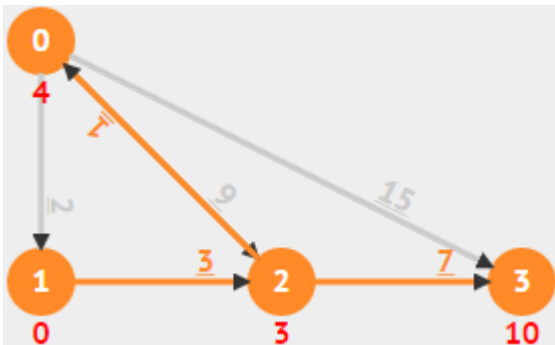
A sample Singapore map; Create and view this directed weighted graph at [VisuAlgo](#)

Query(0, 3, 4): If Ket Fah's current position is at vertex $s=0$, the chosen hospital is at vertex $t=3$, and we can use up to $k=4$ junctions, then the quickest way is this path: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ with total estimated traveling time of: $2+3+7 = 12$ minutes.



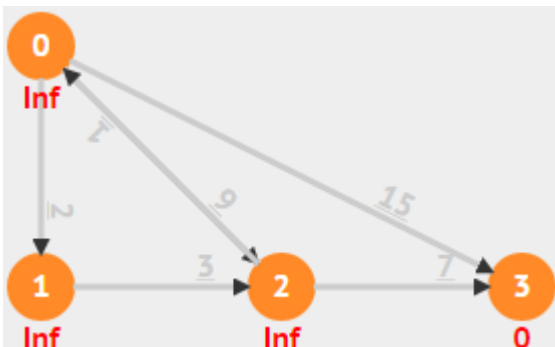
The shortest path from $s=0$ to $t=3$ with no more than $k=4$ junctions in this sample Singapore map

Query(1, 0, 4): If Ket Fah's current position is at vertex $s=1$, the chosen hospital is at vertex $t=0$, and we can use up to $k=4$ junctions, then the quickest way is this path: $1 \rightarrow 2 \rightarrow 0$ with total estimated traveling time of: $3+1 = 4$ minutes.



The shortest path from $s=1$ to $t=0$ with no more than $k=4$ junctions in this sample Singapore map

Query(3, 2, 4): If Ket Fah's current position is at vertex $s=3$, the chosen hospital is at vertex $t=2$, and we can use up to $k=4$ junctions, then there is no path possible, so we output -1 as our answer.

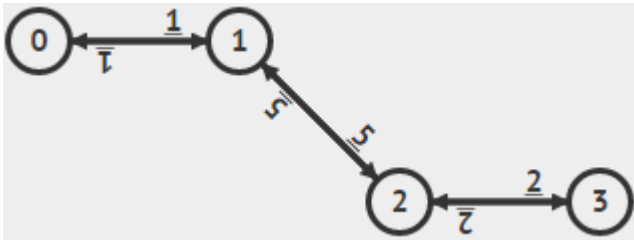


There is not path from $s=3$ to $t=2$ with no more than $k=4$ junctions in this sample Singapore map

Subtask A Constraints (40 points)

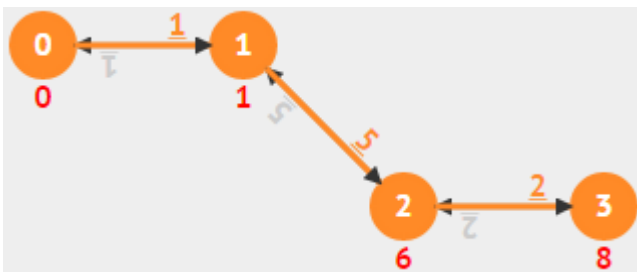
Time Limit: 2s.

On a rather 'impossible case' that simplifies this problem, the road network in Singapore is an **undirected weighted tree**, $1 \leq V \leq 1\,000$ (you can infer E from V), $1 \leq Q \leq 100\,000$, $0 \leq s, t < V$, and $k=V$. We guarantee that $\text{weight}(u, v) = \text{weight}(v, u)$ for all vertices u and v in the graph.



A simplified Singapore Map (Tree); For your convenience, view this undirected weighted tree at [VisuAlgo](#); This is the visualization of the first test case in the Sample Input

Query(0, 3, 4): If Ket Fah's current position is at vertex $s=0$, the chosen hospital is at vertex $t=3$, and we can use up to $k=4$ junctions, then the quickest way is this path: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ with total estimated traveling time of: $1+5+2 = 8$ minutes.



The shortest path from $s=0$ to $t=3$ with no more than $k=4$ junctions in this simplified Singapore map (Tree)

Query(0, 2, 4): If Ket Fah's current position is at vertex $s=0$ and the chosen hospital is at vertex $t=2$, and we can use up to $k=4$ junctions, then the quickest way is this path: $0 \rightarrow 1 \rightarrow 2$ with total estimated traveling time of: $1+5 = 6$ minutes. The visualization of this path is similar as above.

Hint: Have you seen a PS with insane number of queries like this before?

Sample Input

```

2
4
1 1 1
2 0 1 2 5
2 1 5 3 2
1 2 2
4
0 3 4
0 2 4
0 1 4
0 0 4

2
1 1 7
1 0 7
4
0 0 2
0 1 2
1 0 2
1 1 2
  
```

Sample Output

```

8
6
1
  
```

```
0
0
7
7
0
```

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](#).

Please use [BleedingVerifierA.java](#) (click to view) to verify whether your custom-made test data conform with the required specifications.

Problem Author

Dr Steven Halim/Dr Chong Ket Fah
For CS2010.

Footnotes

¹To simplify this problem, let's assume that this time estimation is accurate and there is no traffic jam in Singapore.

²The reason of this constraint **k** is because when bleeding starts, you don't want to move the affected joint as much as possible. However with many junctions there will be much starting and stopping of the car and this will affect the injured joint. Thus Ket Fah want the taxi drive or his brother to cross *no more than k* junctions (vertices in Singapore map) along the shortest path from vertex **s** to vertex **t**.

³Yes, Singapore map does not looks like this, but let's just assume it is.

⁴If 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.

CodeCrunch

[Home](#) | [My Courses](#) | [Browse Tutorials](#) | [Browse Tasks](#) | [Search](#) | [My Submissions](#) | [Logout](#) | Logged in as: **e0009011****CS2010 2016/2017 Sem2: PS5 B****Tags & Categories**

Tags:

Categories:

Related Tutorials**Task Content**

A bleeding episode, v2017 (Subtask B)

The Actual Problem Description

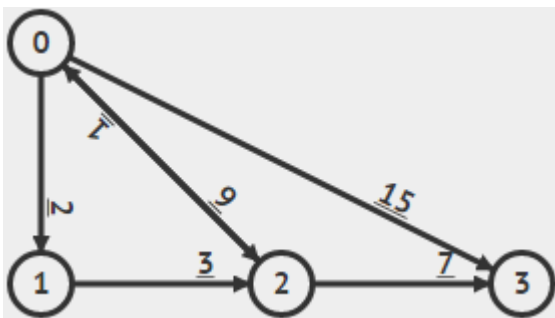
Please refer to Subtask A for the full problem description.

Subtask B Constraints (additional 30 points)

Time Limit: 1s.

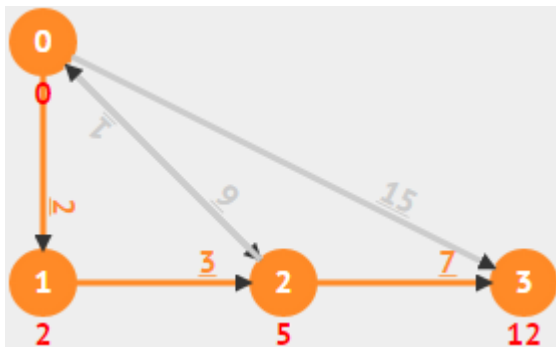
The road network in Singapore is a **directed weighted graph**, $1 \leq V \leq 1\,000$, $0 \leq E \leq 200\,000$, $1 \leq Q \leq 10\,000$, $0 \leq s \leq 9$, $0 \leq t < V$, and $k=V$.

In fact, the first examples shown earlier in Subtask A and replicated below fits this description (see the first graph and the first set of queries in Sample Input/Output).



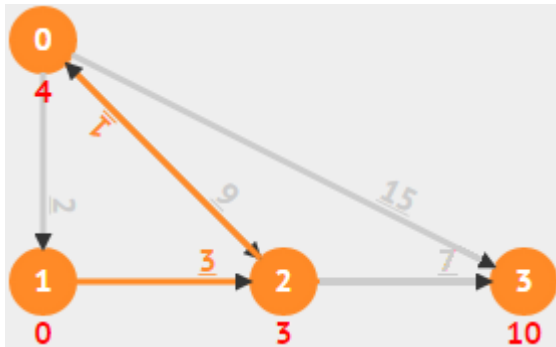
A sample Singapore map; Create and view this directed weighted graph at [VisuAlgo](#)

Query(0, 3, 4): If Ket Fah's current position is at vertex $s=0$, the chosen hospital is at vertex $t=3$, and we can use up to $k=4$ junctions, then the quickest way is this path: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ with total estimated traveling time of: $2+3+7 = 12$ minutes.



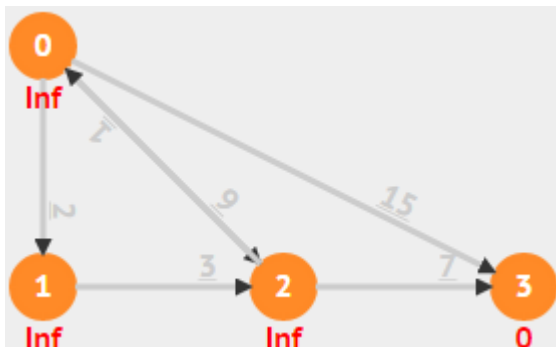
The shortest path from $s=0$ to $t=3$ with no more than $k=4$ junctions in this sample Singapore map

Query(1, 0, 4): If Ket Fah's current position is at vertex $s=1$, the chosen hospital is at vertex $t=0$, and we can use up to $k=4$ junctions, then the quickest way is this path: $1 \rightarrow 2 \rightarrow 0$ with total estimated traveling time of: $3+1 = 4$ minutes.



The shortest path from $s=1$ to $t=0$ with no more than $k=4$ junctions in this sample Singapore map

Query(3, 2, 4): If Ket Fah's current position is at vertex $s=3$, the chosen hospital is at vertex $t=2$, and we can use up to $k=4$ junctions, then there is no path possible, so we output -1 as our answer.



There is not path from $s=3$ to $t=2$ with no more than $k=4$ junctions in this sample Singapore map

Hint: Have you seen a PS with insane number of queries like this before?

Sample Input

```
2
4
3 1 2 2 9 3 15
1 2 3
2 0 1 3 7
0
3
0 3 4
1 0 4
3 2 4

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

```
4
0 3 4
0 2 4
0 1 4
0 0 4
```

Sample Output

```
12
4
-1

8
6
1
0
```

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](#).

Please use [BleedingVerifierB.java](#) (click to view) to verify whether your custom-made test data conform with the required specifications.

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

[Home](#) | [My Courses](#) | [Browse Tutorials](#) | [Browse Tasks](#) | [Search](#) | [My Submissions](#) | [Logout](#) | Logged in as: **e0009011**

CS2010 2016/2017 Sem2: PS5 C

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

A bleeding episode, v2017 (Subtask C)

The Actual Problem Description

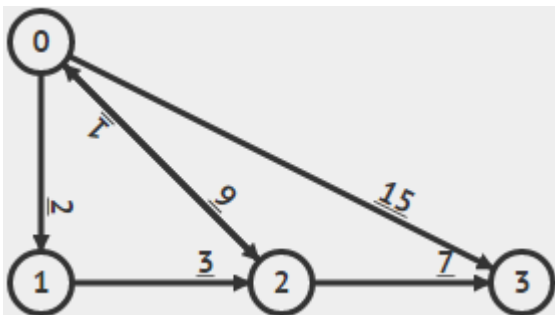
Please refer to Subtask A for the full problem description.

Subtask C Constraints (additional 30 points)

Time Limit: 1s.

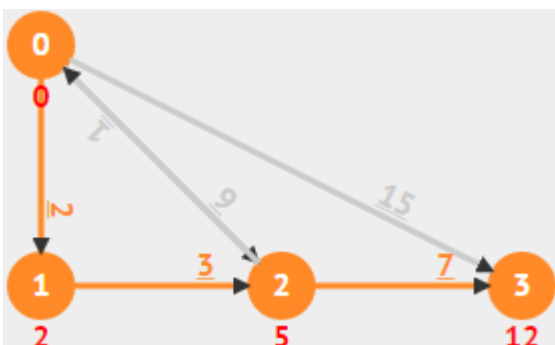
The road network in Singapore is a directed weighted graph, $2 \leq V \leq 1\,000$, $0 \leq E \leq 200\,000$, $1 \leq Q \leq 20$, $0 \leq s, t < V$, $1 \leq k \leq \min(V, 20)$, and $s \neq t$.

Let's take a relook at the sample graph that has been shown in Subtask A earlier:



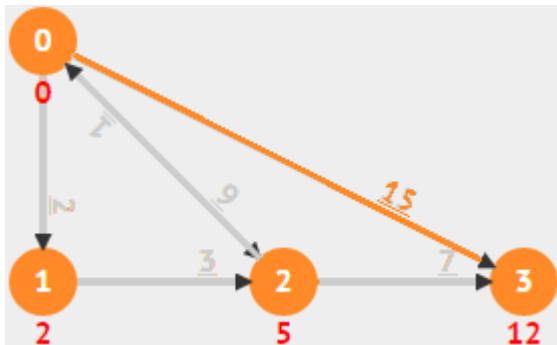
A sample Singapore map; Create and view this directed weighted graph at [VisuAlgo](#)

Now recall that the answer for Query(0, 3, 4) is: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ with total estimated traveling time of: $2+3+7 = 12$ minutes. Notice that this path uses 4 junctions (vertices).



The shortest path from $s=0$ to $t=3$ with no more than $k=4$ junctions in this sample Singapore map

Now if we have Query($0, 3, 3$), then path: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ shown above is *invalid* as it uses 4 junctions. The valid shortest path from $s=0$ to $t=3$ that uses no more than $k=3$ junctions is path: $0 \rightarrow 3$ with total estimated traveling time of: 15 minutes. This path is 3 minutes longer than the true shortest path without Ket Fah's restriction of not crossing more than k junctions but it is now the best answer¹ for this Subtask C.



The *new* valid shortest path from $s=0$ to $t=3$ with no more than $k=3$ junctions in this sample Singapore map

Sample Input

```
1
4
3 1 2 2 9 3 15
1 2 3
2 0 1 3 7
0
12
0 3 4
0 3 3
0 3 2
0 3 1
1 0 4
1 0 3
1 0 2
1 0 1
3 2 4
3 2 3
3 2 2
3 2 1
```

Sample Output

```
12
15
15
-1
4
4
-1
-1
-1
-1
-1
-1
-1
```

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](#).

Please use [BleedingVerifierC.java](#) (click to view) to verify whether your custom-made test data conform with the required specifications.

Problem Author

Dr Steven Halim/Dr Chong Ket Fah
For CS2010.

Footnotes

¹There is one other possible path from **s=0** to **t=3** that uses no more than **k=3** junctions, path: 0 → 2 → 3, but it has longer estimated traveling time of: 9+7 = 16 minutes.

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.

© Copyright 2009-2017 National University of Singapore. All Rights Reserved.
[Terms of Use](#) | [Privacy](#) | [Non-discrimination](#)

[MySoC](#) | [Computing Facilities](#) | [Search](#) | [Campus Map](#)
School of Computing, National University of Singapore