

Lab Demo 09

Friday, 02 Novemeber 2018 – Tomorrow is PE Day!!

PE Practice 2 A/B

- Preprocessing step: stores all N names into a `std::set` (no duplicates handled by it), erase all M tabooed names from the set ($O(M\log(N)) + O(N\log(N))$)
- Go through each pair (transfer to a vector first), get all concatenated names where 1st name starts with 'J' ($O(N^2)$)
- A: sort all names (`std::sort`) according to condition (largest length, alphabetically order if tie)
- Can use a `pair<int,string>`, 1st element is negated length of name, 2nd element is name itself

PE Practice 2 A/B

- B: brute force sort which is $O(X \log(X))$ may be too slow (but ACed for me =O), where $X = N^2 \approx 2.5\text{Mil}$
- Note that we only want the 1st 100 elements
- Can create a ``std::priority_queue`` with its constructor in $O(X)$ time, pop 100 times.
- Use ``std::algorithm`` ``nth_element`` at 100 position, sort first 100 elements
- Use ``std::algorithm`` ``partial_sort`` for 1st 100 elements (same as PQ method)
- Time complexity: $O(X) + O(K \log(X))$, where K is a constant 100

PE Practice 2 C/D/E

- C/D: go through each line 1 by 1, if `SINGAPORE` or `0` appears, add the other element answer (vector/set), sort values before printing
- E: for each pair of elements, map them to their neighbor city `map<string, set<string>>`
- If `SINGAPORE` is in the pair, add all neighbours to `SINGAPORE` to an answer set, and another set `len1` to indicate cities 1 length from `SINGAPORE`. Also, add all other neighbours to that other city to the answer set.
- For each other pairs which don't have `SINGAPORE`, check if either elements are inside the `len1` set, if they are, add the other 1
- Can also solve using BFS to find all neighbours length 2 from `SINGAPORE`, or modified DFS (graph won't be tested for this sem PE though)

Graph DS Review

- VA: <https://visualgo.net/en/graphds>
- Code: ch2_07_graph_ds.cpp from CP3 book: <http://cpbook.net/#downloads>
- DFS/BFS example

PS5 Discussion, Subtask A+B

Big Question: Does parameter **k** matters?

PS5 Subtask A+B Summary

Nothing to hide here, parameter **k** does not matter for these two subtasks...

PS5 Subtask A and B are there to force you to code **at least one** SSSP algorithm:

- Subtask A \rightarrow the graph is a **tree**, $V < 1K$, $Q \leq 100K$
therefore the best algorithm is _____DFS/BFS_____
- Subtask B \rightarrow the graph is **weighted**, $V+E < 201K$, $Q \leq 10K$
therefore the best algorithm is _____Dijkstra_____

But how to deal with that big **Q**?

Dijkstra's is a very standard algorithm,
but to avoid (minimize) plagiarism check...

- Choice of Dijkstra's version
 - Original or Modified **(2 options)**
- Choice of Priority Queue implementation for Dijkstra's
 - TreeSet, TreeMap :O, PriorityQueue (Lazy DS), your own PS4 code (AVL Tree), your own PS3 code (Binary Heap), or even Fibonacci Heap (or other exotic Heap DS)... **(6 options)**
- Or 1 other alternative outside Dijkstra's: SPFA **(+1 option)**
- Technically, we have **$2*6+1 = 13$ combinations** 😊

PS5 Subtask C

Why it is hard(er)?

The additional constraint is quite relevant in real life

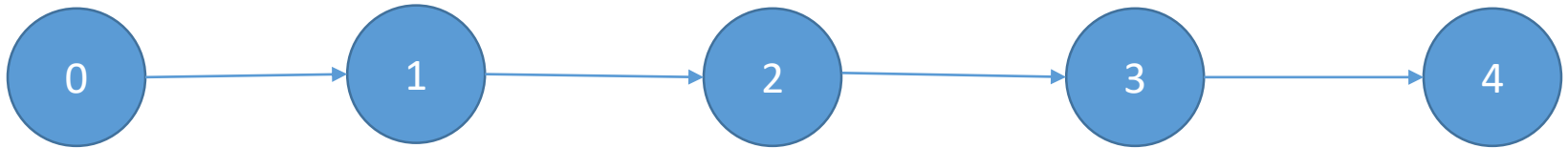
- Google around for recent (taxi) accidents videos in SG
 - Many happened around junctions
- Note that some of those accidents are **fatal :S:S:S** so please watch those videos with caution

What to do in order to handle this seemingly simple additional constraint that the shortest path cannot have more than **k** vertices on it?

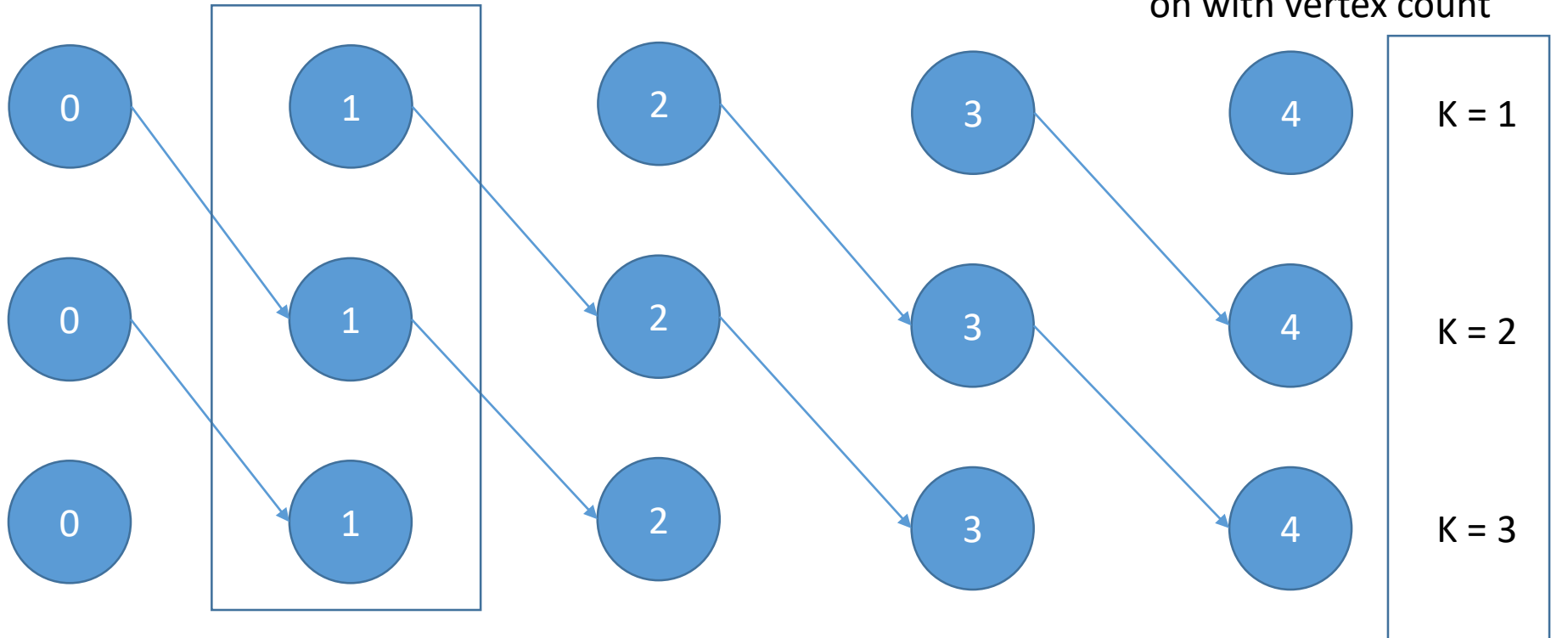
- Hint: Put that vertex usage information on each vertex...

Graph Modelling

Original Graph

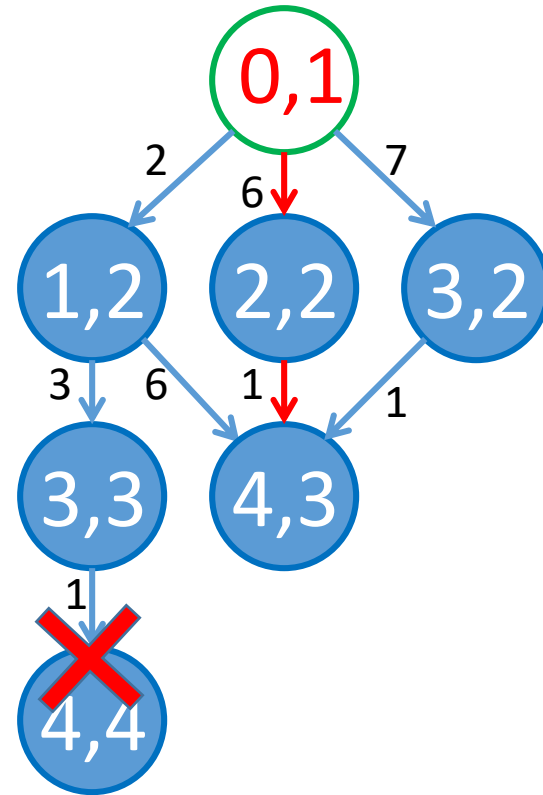
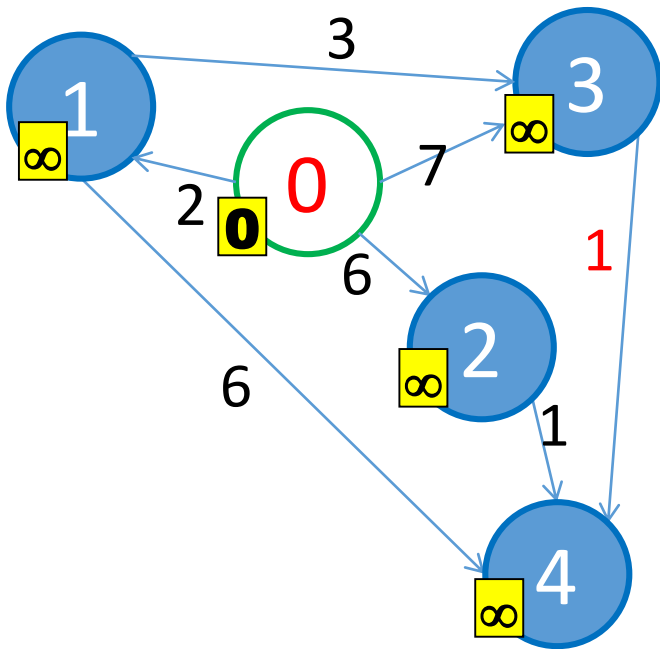


Transformed Graph



Restrict SP to 3 vertices only

The transformed graph is a DAG!



Note that if I do not restrict the number of vertices in the SP, the answer is $0 \rightarrow 1 \rightarrow 3 \rightarrow 4$ of weight $2+3+1 = 6$ instead of 7 from path $0 \rightarrow 2 \rightarrow 4$

VisuAlgo Online Quiz is Next Week!

VisuAlgo Online Quiz is done during the Lab Demo 10 slot itself *(just to prove that you are the one doing the quiz, not someone else)*

Online Quiz covers **all stuffs** in CS2040C: Sorting, Linked List, Binary Heap, Hash Table, BST, AVL Tree, Graph DS, DFS/BFS, SSSP

Man versus Machine (1)

How to ace VisuAlgo Online Quiz?

Answer: Simple, **practice like crazy**

- VisuAlgo Online Quiz is just a collection of (PHP+MySQL) scripts at the server side and (JS) scripts at the client side
 - Unless Steven sets up new (hard) questions which will only appear during the actual Online Quiz, Steven won't be able to change questions that are already in the system easily
 - So, after enough # of trainings, you will start seeing similar question types*, just with random^ parameters...

Man versus Machine (2)

Some questions have shortcuts...

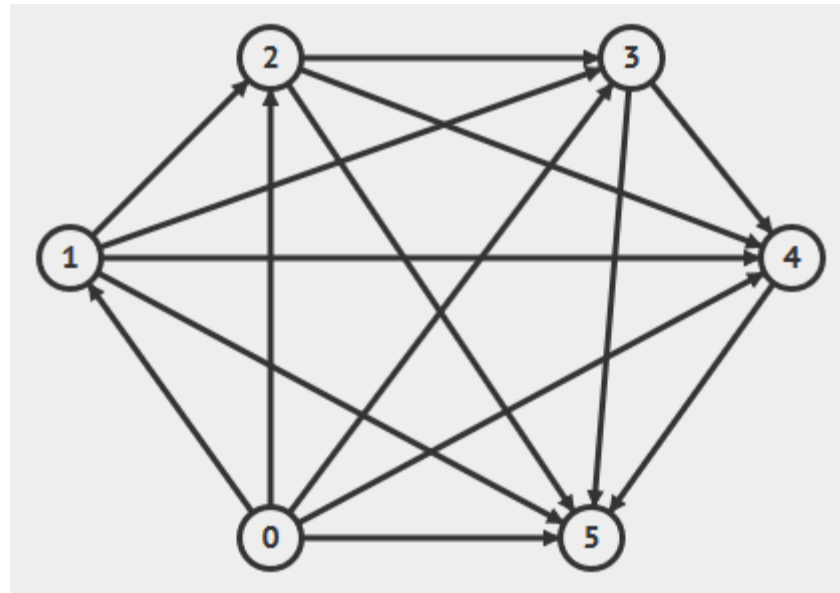
- Example 1: Perform ExtractMax() **k** times from Binary Max Heap of **n** elements, click all elements that are left behind?
 - Naïve: Perform ExtractMax **k** times, click elements that are left behind
 - Clever: Click **n-k** smallest elements in that Binary max Heap :O...
- Example 2: How many structurally different BSTs can you form with <a randomized small number> distinct elements?
 - Woah, what kind of question is this?
 - It was from Written Quiz 1 in S1 AY 2014/2015
 - Super hard if you don't know what to do, but very easy otherwise if you can think (or find solution a.k.a. Googling) **outside** the test time
 - The answer is **Catalan Numbers**

Man versus Machine (3)

- Example 3: What is the minimum number of vertices in an AVL tree of <a randomized small number> height?
 - Another question that is solvable by using a formula
 - Hint: an AVL tree of height **h** requires two subtrees of heights **$h-1$** and **$h-2$**

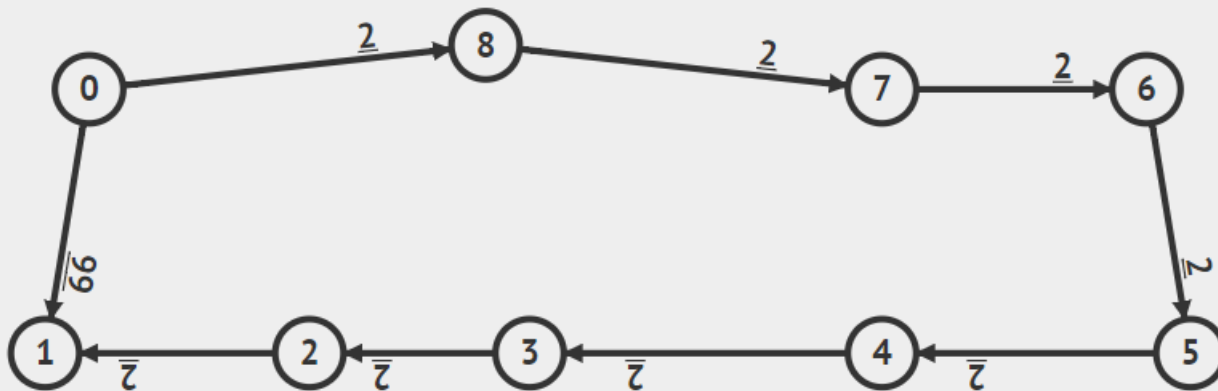
Man versus Machine (4)

- Example 4: Draw a DAG with **V** vertices and **E** edges
 - How to keep drawing neat when **E** is large?
 - Draw a regular polygon, only draw edges from smaller vertex to larger vertex
 - Eg for **V** = 6, **E** = 15 (maximum number of edges)



Man versus Machine (4)

- Example 5: Draw graph such that Bellman-Ford runs at least X passes:
 - Draw edges from source to (V-1) edge, (V-1) to (V-2), and so on
 - Only 1 vertex updated at each pass, guaranteed to run maximum passes



Show Off

Let's try the training mode of CS2040C OQ topics

- I will do one round of show-off to motivate you all to do the same thing in your official Online Quiz so that your basic understanding of these topics are top notch 😊
- Watch and analyze me as I do a **speed run** (10 hard questions in only 5 minutes)

<https://visualgo.net/training?diff=Hard&n=7&tl=5&module=sorting,list,heap,hashtable,bst,avl,graphds,dfsbfss,sss>

All the best for tomorrow...

If you want more exercises

- Just ask your Lab TA on what topic that you want to practice on
 - Lab TA will then show an online judge exercise on the topic with the highest demand in that lab group, and as usual, do live solving (this time without waiting for students as we may run out of time otherwise)
- Come to your assigned PL by 12.50pm to login to desktop and ensure you are not late for PE