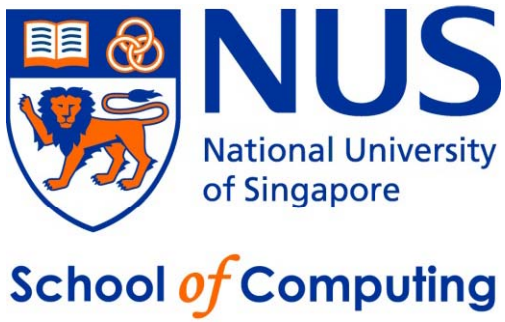
CS2010 – Data Structures and Algorithms II

Lecture 12 – Mystery Lecture

stevenhalim@gmail.com



Admins

- Appeal period for CS2010 Quiz 2
  - Latest by **Thursday, 8 Nov 2012, 7pm**
  - Compare your marks with the modal answer + marking scheme + statistics file in IVLE Workbin
  - If you think you should have gotten more marks based on the wording of the marking scheme, contact us (Ket Fah for Q1-4, Steven for Q5)

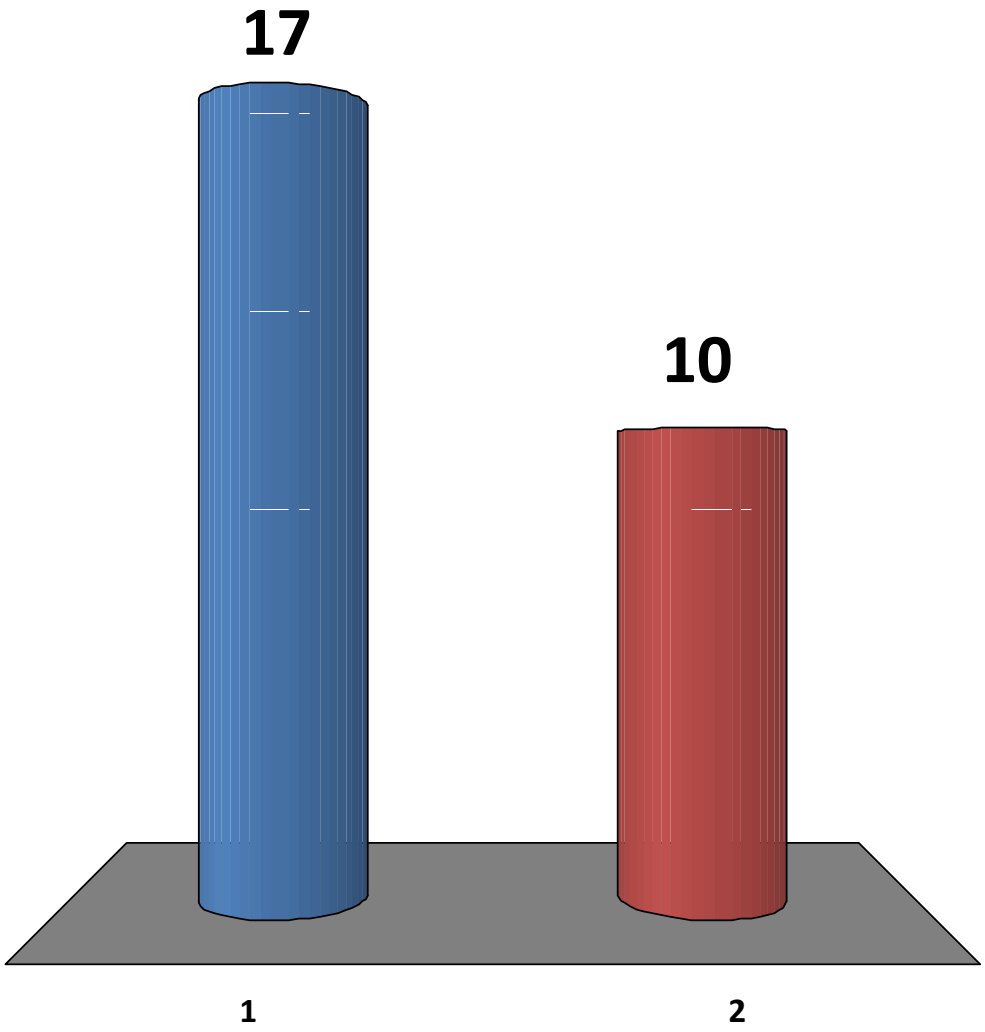
Outline of the Last Mystery Lecture

- What are we going to learn today?
  - Something mysterious :D
  - Review of CS2010
  - Some parting words
    - Reflection of the current CS2010
    - Modules in SoC beyond CS2010
    - Final Exam Tips
    - Part time TA jobs
    - Admin: Please **return** your clicker!

Do you bring your clicker?

0 marks for your CS2010 final exam if you do not/cannot click yes ☹

1. Yes
2. No (do NOT click me)



Important

- For those who click “yes” (or “no” ☺)
  - You will need to return your clicker today
  - We will do this **later** at the end of this lecture
- For those who cannot click…
  - You will **not** get 0 marks for final if you:
    - Meet Steven during reading week and return your clicker (if you just forgot to bring it today)
      - PS: Steven is away to Hat Yai for ACM ICPC during Week13
    - Or pay 48.15 SGD replacement fee if you really lost / broke it …
      - Run out of battery is “ok”

THAT MYSTERIOUS TOPIC IS…
MCBM

QUICK SUMMARY OF CS2010

Two Non-Linear Data Structures

- L1: Introduction, problem solving techniques
- L2: ADT Table, Binary Tree, Binary Search Tree (BST)
- L3: The Concept of Balanced BST, AVL Tree
- **PS1/R: Baby Names**
- L4: ADT Priority Queue, Binary Heap, Compact Array
- **PS2/R: Scheduling Deliveries**

Graph DS and Traversal

- L5: Graph DS: AdjMatrix, AdjList
- L5: Graph Traversal: BFS/DFS and its applications Reachability, Connected Components, Toposort
- L6: Additional Graph DS: EdgeList
- L10: Special graph that has 2 edges/vertex
- Implicit Graph
- The notion of **Space** Complexity
- Benefits/Disadvantages of using them
- Conversion between one graph DS to another
- **PS3/R: Hospital Tour**

MST

- L6: Kruskal's Algorithm
- L6: Union Find Data Structure (Not examinable)
- L6: Prim's Algorithm
- L6: Heap Data Structure again
- Technique of modeling problem as an MST problem
- **PS4: Out for a Walk**

SSSP and Various Special Cases

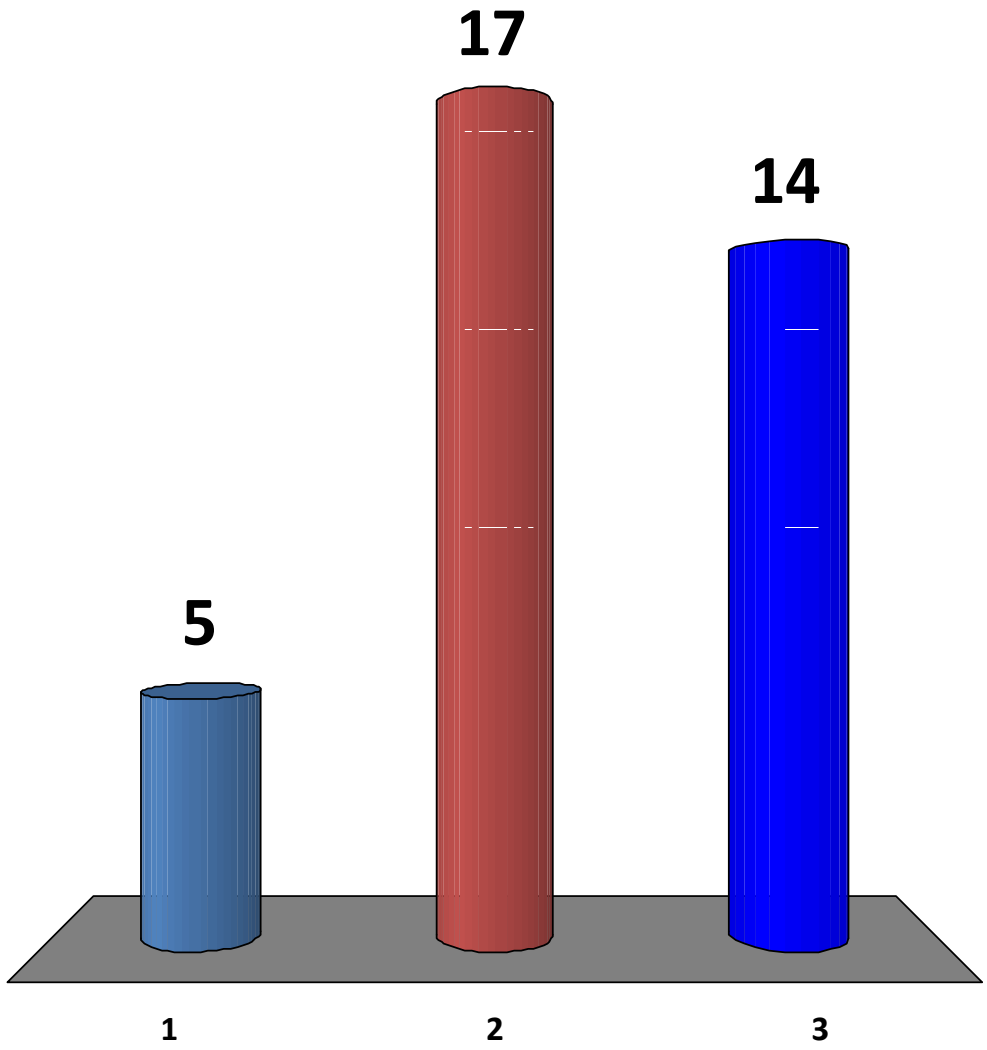
- L7: General Shortest Paths Algorithm: $O(\infty)$
- L7: Bellman Ford's: O(VE) – general
- L7: BFS: O(V + E), unweighted graph
- L8: Dijkstra's: O((V + E) log V) – binary heap
  Original versus Modified implementation
- L8: DFS/BFS: O(V), Tree
- L8: Toposort + DP: O(V + E), DAG
- Various occasions: SSSP Problem Modeling
- **PS Bonus, PS5, Quiz 2 Section 5**

DP (have I test you on this?)

- L9: Algorithms on DAG
  - One-pass Bellman-Ford's is a nickname for SSSP on DAG solution
- L10: Algorithms on (Implicit) DAG
  - Adding (for CS2010, at most) one parameter to convert general graph into a DAG
- L11: APSP: Floyd Warshall's DP algorithm
- Space Complexity: Vertices in implicit DAG/states
- Time Complexity: Edges in implicit DAG/transitions
- Various occasions: DP problem modeling
  - Top-down versus Bottom-up implementation
- **PS6, PS7/R, PS8, final exam question?**

The third part of CS2010 (DAG/DP)

1. I am OK
2. I am behind but will catch up during reading week
3. I am in danger…



SOME PARTING WORDS

Things That Can Be Done Better

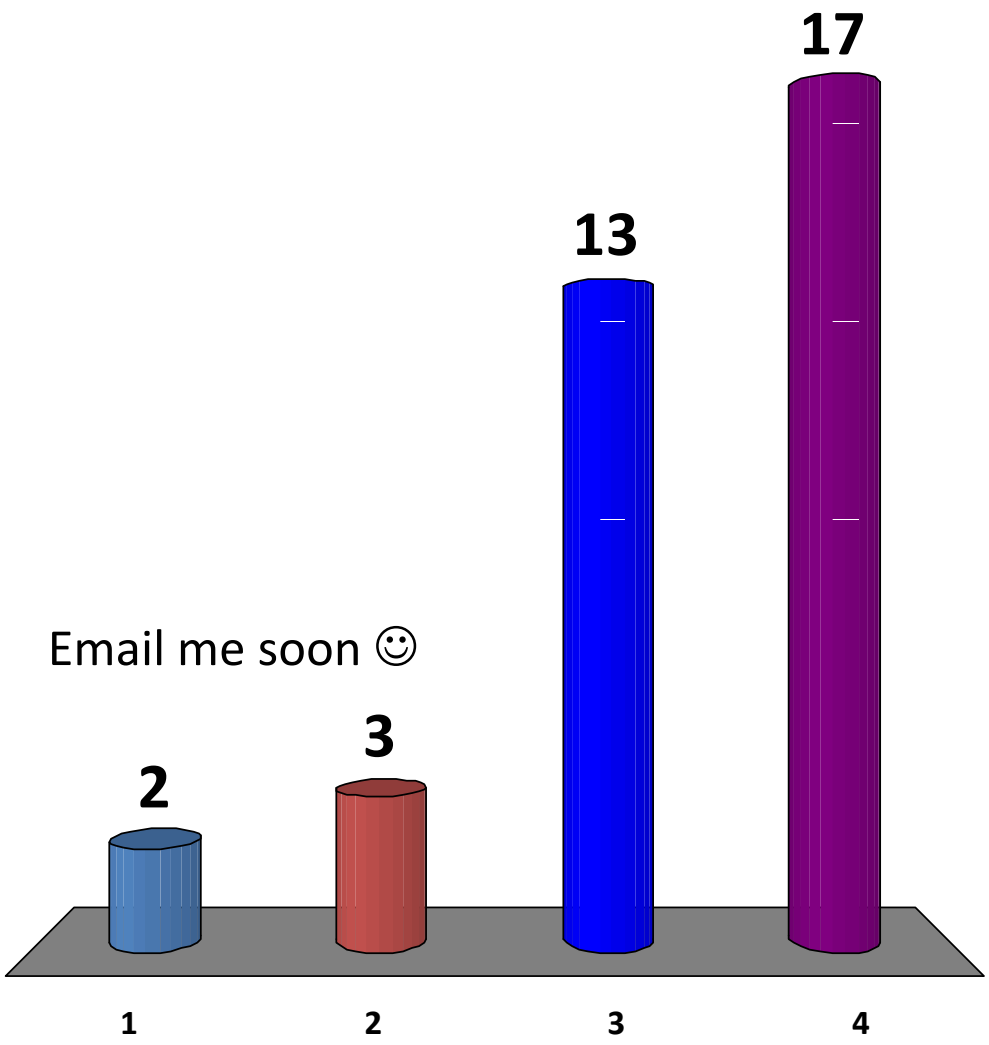
- SoC won't have to run e-Learning week next year (2013); this should improve Week09 experience
  - I hope the "graph modeling" skill can be transferred in a better way as this is an important skill for CS2010
- Leftover from last year:
  - Further reduce the usage of global variables in demos
  - Use more OO examples, e.g. Floyd Warshall's demo
- Discuss inheritance in BST to AVL code (leftover from CS1020?)
- Further reduce the already minimal typos/mistakes in lecture notes/PSes, integrate the R-experience better
- Etc (we will look at your NUS Online Teaching Feedback regarding this module ☺)

Relevant Modules After CS2010 (1)

- CS3233: Competitive Programming
  - Pre-req: **at least** A- from CS2010/20, **(very)** strong interest in algorithms, programming (especially implementation), and problem solving (2-3 PSes *worth of work* **every week**!)
  - Must free-up your schedule every Wednesday 6-9pm on the 2nd semester…
  - CS3233 syllabus will further evolve because of CS2010/20
- CS3230: Design and Analysis of Algorithms
  - Discussion: By now, "the information propagation" should be completed
- CS5206: Foundations in Algorithms

CS3233: Competitive Programming

1. I will **definitely** take your challenging module next sem (S2, AY2012-2013)
2. I will consider ☺, maybe I will take it…
3. I will consider, but at the moment I do not have plan to take it…
4. I know for sure that this module is not for me, thanks



Relevant Modules After CS2010 (2)

- Algorithms/Techniques on parallel machines:
  - CS3210: Parallel Computing
  - CS3211: Parallel and Concurrent Programming
  - CS4231: Parallel and Distributed Algorithms
- Software Engineering:
  - CS2103: Software Engineering
  - CS3215: Software Engineering Project
- CS3243: Artificial Intelligence

Final Exam (1)

- An open book test, but here is my suggestion
- Prepare a “data structure”:
  - Several sheets of A4 paper, as many as you need…
    - But not too many :O
  - Either handwritten or printed
  - Use whatever font size that is convenient for you
- Purpose:
  - To improve your search performance during exam time from O(N) *where N is the number of pages of books or lecture notes that you bring into exam hall* into O(1) where you can use a kind of “hashing” to identify the required information in your A4 paper(s)…

Final Exam (2)

- There will be 4 questions (OK, 4 question **types**)
  - Not of the same length
  - Not of the same marks
  - Mostly, similar to our Quiz 1 and Quiz 2 styles
  - Topics: From the entire semester **with focus** on things that are not yet tested in Quiz 1 and Quiz 2 :O
    - Routine questions in front, as time differentiator, they are *basics*
    - Some analytical questions, also as time differentiator
    - 2 short/medium 'new graph theory' problems like MCBM that worth ~one letter grade per question, e.g. B+ to A-, A- to A, etc…
    - 1 long DP? question that potentially worth ~two letter grades :O, e.g. B to A-, B+ to A, etc…

Remarks from Quiz 2

- Let's see quiz2 solution file, page 14…
  - Remarks about Question 3 & 4 statistics from Quiz 2 and their relevance with Question 2 & 3 of Final Exam
  - Avoid repeating similar errors in final exam

Final Exam (3)

- Time Management
  - You have 120 minutes as with Quiz 2
  - I *assume* that you have O(1) searching performance with help of your few pages of A4 sheets for Q1 & Q2 ☺
    - This kind of O(1) searching performance is needed to clear the earlier "time differentiator" questions as fast as possible…
  - I believe you all will have time to read all problems
    - Note: Do not skip the sub-questions in each question, they are there to guide you!
  - But, I estimate that you will likely spend more time *thinking* than writing the (*short*?) answers

Final Exam (4)

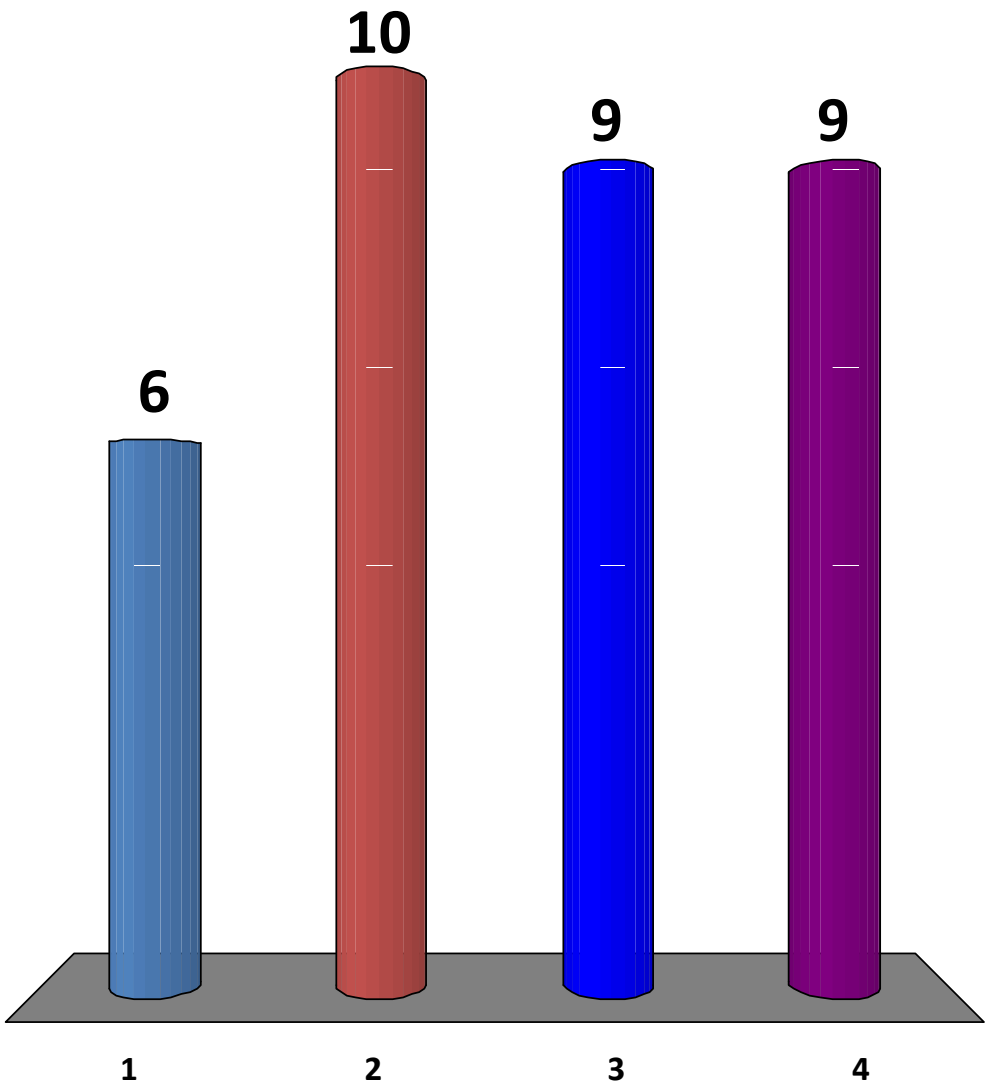
- Study Preparation
  - In case you are not aware, few years ago I have compiled solutions for a lot of past exam papers of the (old) CS1102
    - http://www.comp.nus.edu.sg/~stevenha/myteaching/exam_hints.pdf
    - But the syllabus of CS2010 is now different, especially my version
      - From Lecture09 → You have to understand how to apply SP/LP/CP on DAG
      - From Lecture10 → How can you transform the given non DAG to DAG by adding at most one parameter?
      - From Lecture11 → Various applications of Floyd Warshall's algorithm
      - From Lecture12 → Creativity on handling "new definitions"
  - Solving some relevant online judge problems related to CS2010 syllabus may help
    - http://uhunt.felix-halim.net/id/32900

Consultation Slots Before Final Exam

- Dr Steven Halim, COM2-3-37
  - Saturday, 10 Nov 2012, 10am-12pm, SR next to LT19
  - Tuesday, 20 Nov 2012, 12-2pm, SR next to LT19
    - Both are informal, casual, not recorded, free and easy
- Dr Chong Ket Fah
  - Email him first
- Any other teaching staffs of CS2010
  - They are all undergrads and have exams too
  - Thus, please redirect questions to myself or Ket Fah ☺

My Study Plan

1. I will come to Sat 10 Nov 2012, 10am-12pm session only
2. I will come to Tue 20 Nov 2012, 12-2pm session only
3. I will come to both sessions
4. I can prepare for final exam on my own



Part-time Teaching Assistant Job (1)

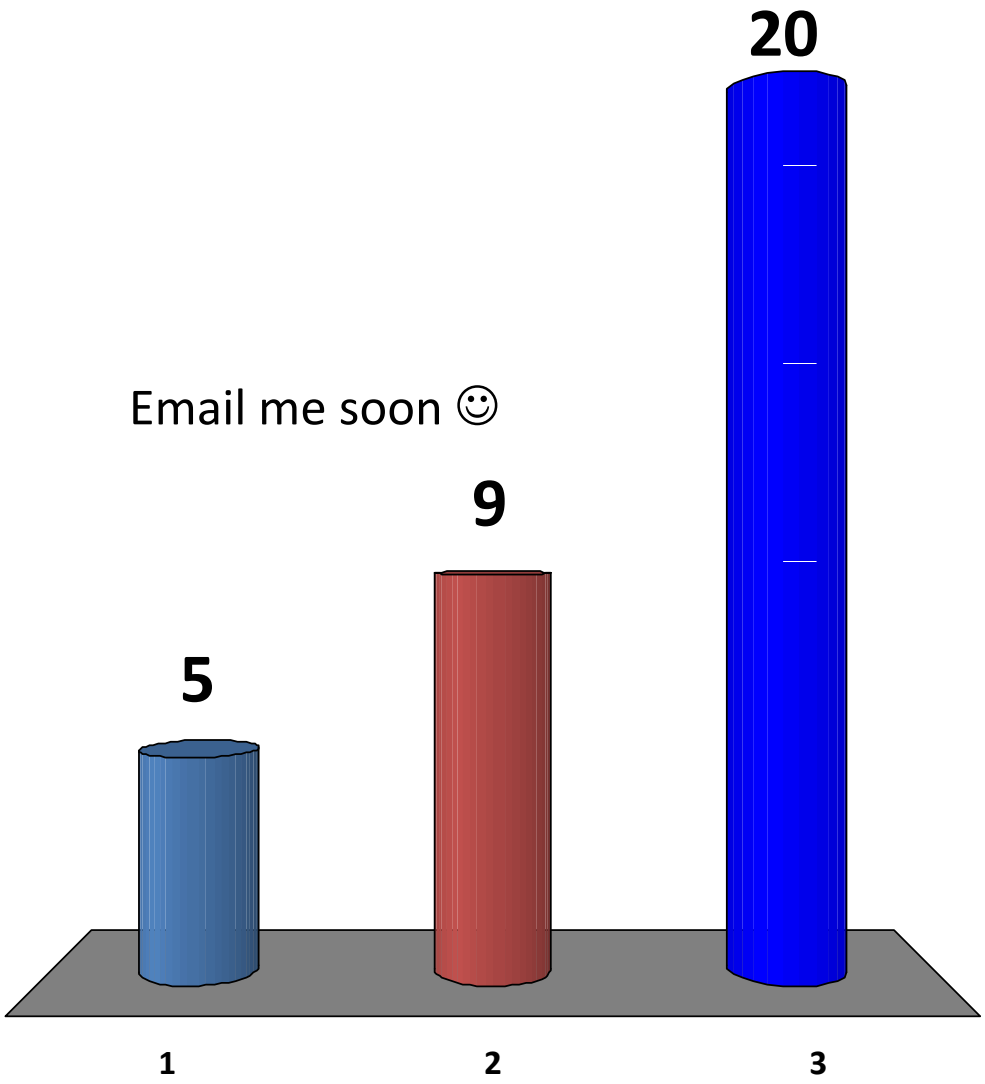
- I will likely teach CS2010 again next year
  - S1, AY2013/2014 (Aug-Nov 2013)
  - Estimated enrollment: ~[120-150] :O
- I want 1:~15 staff:students ratio as with this semester
  - Thus, I need 8-10 staffs, slightly more than this semester
- I prefer TAs who:
  - Like algorithms and problem solving
  - Eventually score at least A- in CS2010
    - You will know this by end of December 2012
  - Like to talk with students (to be precise: your juniors)

Part-time Teaching Assistant Job (2)

- If you are interested, drop me an email as soon as you know your CS2010 grade
  - Or if you are sure that you will get at least A-, then you can email me earlier
  - We will do some interviews, maybe during July 2013
- FYI, Dr Seth Gilbert will teach CS2020 next semester
  - S2, AY2012/2013 (Jan-Apr 2013)
  - Estimated enrollment next year: ~30-40
  - So if you prefer to TA CS2020 next semester instead, drop Seth an email

For Those With Clicker…
Tell me something

1. I am very interested to apply as TA (contact me after lecture)
2. I am interested to apply as TA
3. No thanks



Thank You ☺

- You all (the *third* batch) have just “finished” CS2010
  - Congratulations
- We need your constructive feedback to further improve this module next sem/year
  - CS2010 will be handled by Dr Ang Chuan Heng next semester, I will pass some information to him too
- Please write those comments in your NUS Online Teaching Feedback exercise if you haven’t done so ☺
  - That system is available on Week13-Reading Week

CLICKER!!

- Now, let's return the clicker!
- The procedure:
  - I have sorted the loan forms based on **clicker ID**
  - I will call you per batch:
    1. CIT-I-01 to CIT-I-13,
    2. CIT-I-22 to CIT-I-33,
    3. CIT-I-41 to CIT-I-56, and finally
    4. CIT-I-61 to CIT-I-70
  - Come forward, return your clicker, (I will briefly test it), and I will put a check mark on your name, then you can leave
  - Be patient, this will take time