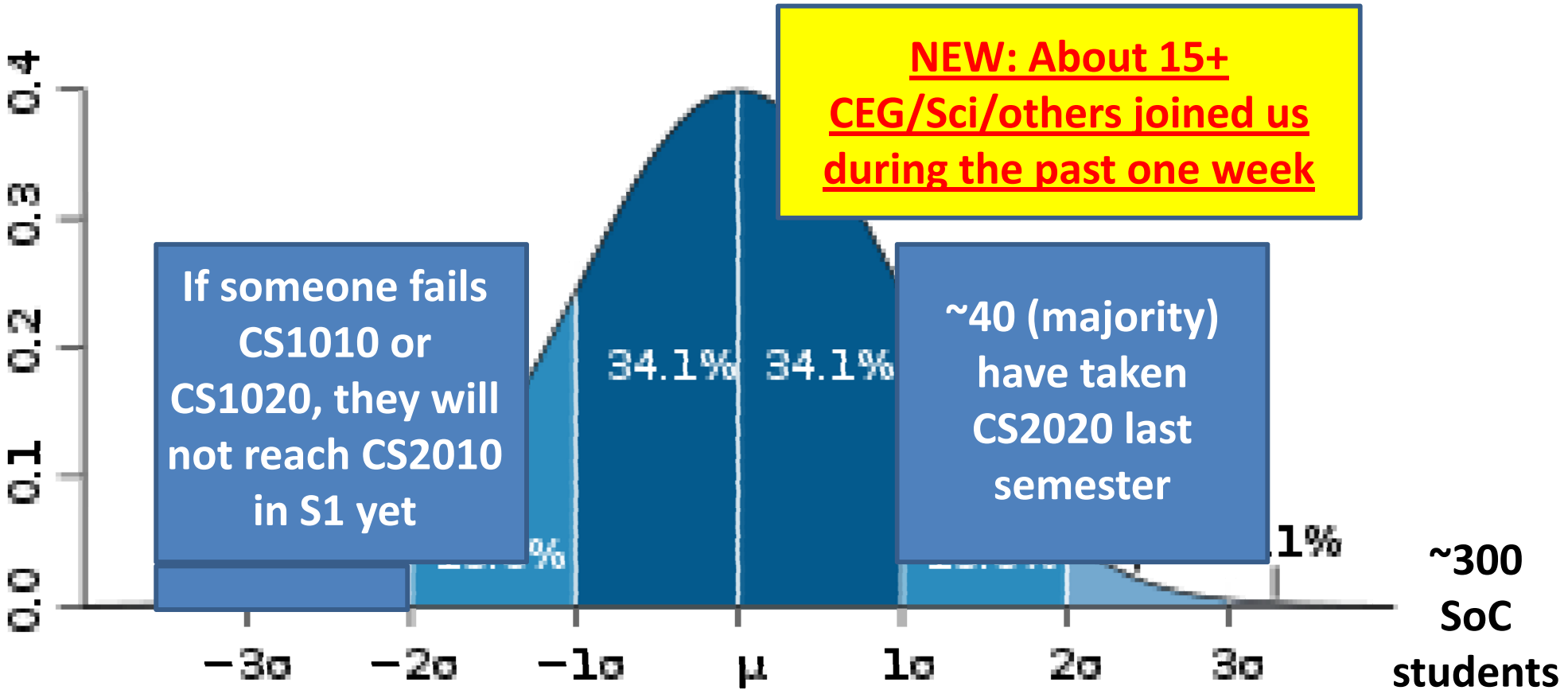
Welcome ☺

- Teaching Staffs Introduction
  - See IVLE Workbin for **a** special PPT
- Class Ratio
  - Tutorial: 10:**142** = 1:~**14+**
    - Class participation is a must! (5%)
  - Lab: 7:**142** = 1:~**20+**
    - Demo attendance (5%)
    - Take home Problem Sets (15%)

Typical Class Profile (CS2010 in S1)

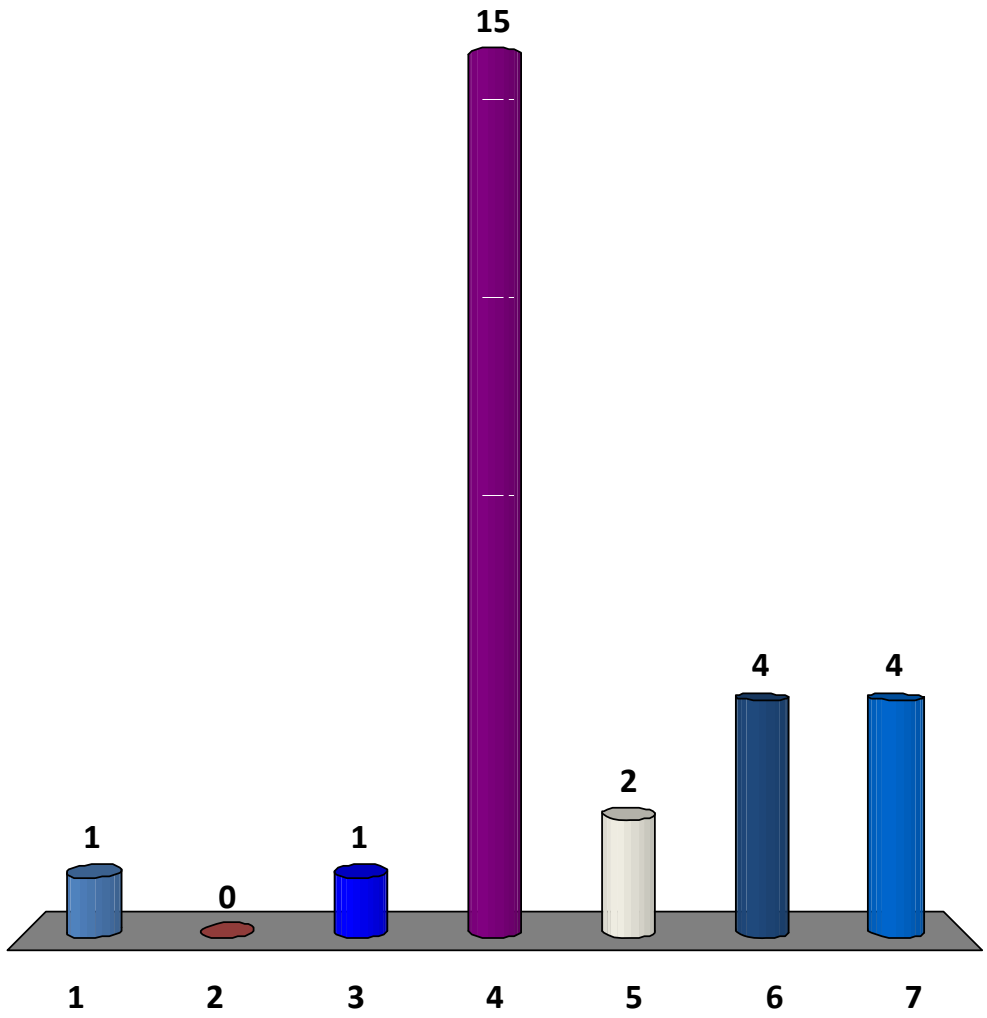


And from these remaining students, about half have chosen IS and do not need to take CS2010 in S1

$f(n) = 5000 \log n * \log n + 5n - n + \text{sqrt}(n^2)$

$f(n) =$

1. $O(n^2)$
2. $O(n \log \log n)$
3. $O(n \log n)$
4. $O(n)$
5. $O(\log \log n)$
6. $O(\log n)$
7. $O(1)$

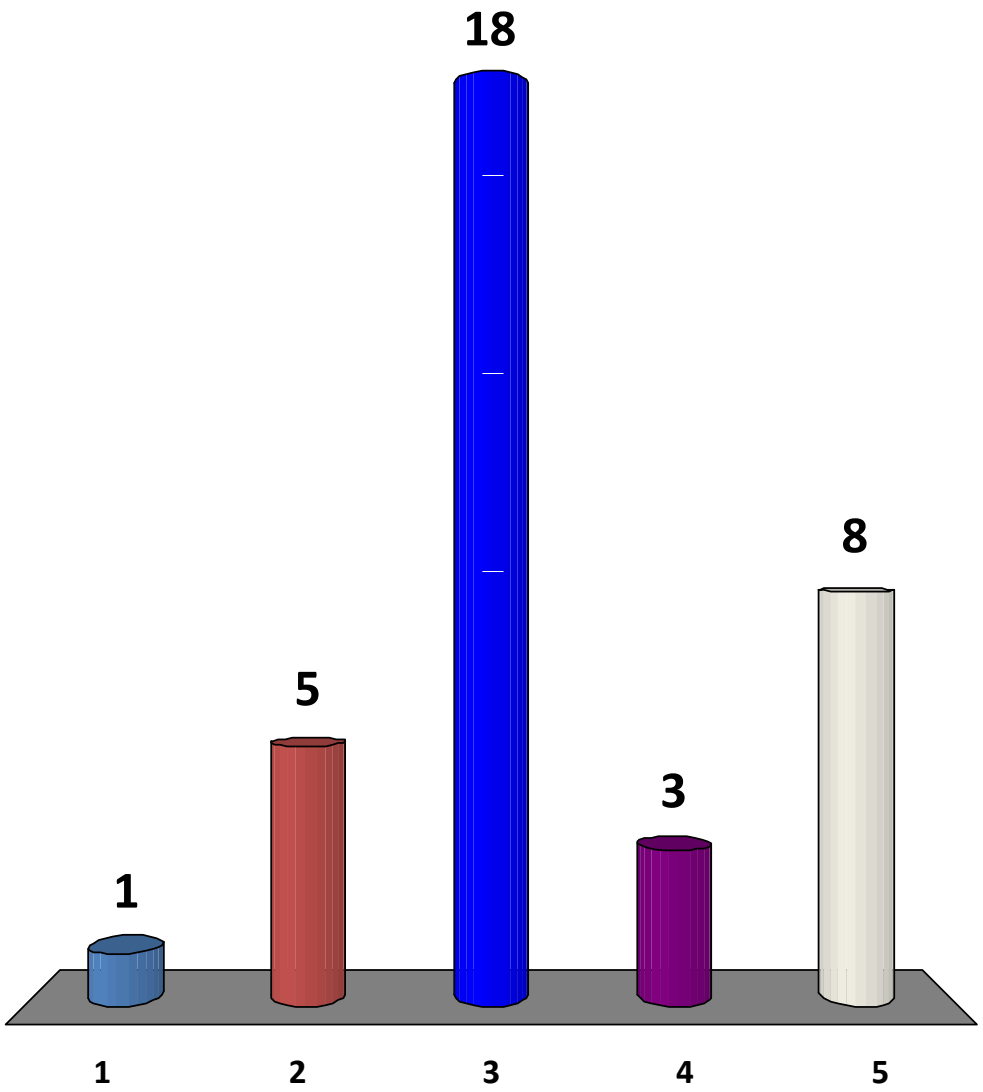


Introducing…

- http://www.comp.nus.edu.sg/~stevenha/visualization
- A **visualization project** between myself, Zi Chun (3rd year SoC), Victor Loh (graduate, work @ FB), Felix (my brother, graduate, work @ Google) and 4 FYP students (Albert, Trang, Peter, Duy)
- A new way of learning data structures & algorithms
  - Explore them ON YOUR OWN!
- Now, if you have either: **iPhone** (or other HTML5 compatible smartphones), **iPad**, or **laptop**, visit that URL and follow me ☺
  - We will start with LinkedList/Stack/Queue visualization
  - http://www.comp.nus.edu.sg/~stevenha/visualization/linkedlist.html

What is the best time complexity to search for an item in an unsorted linked list of size N?

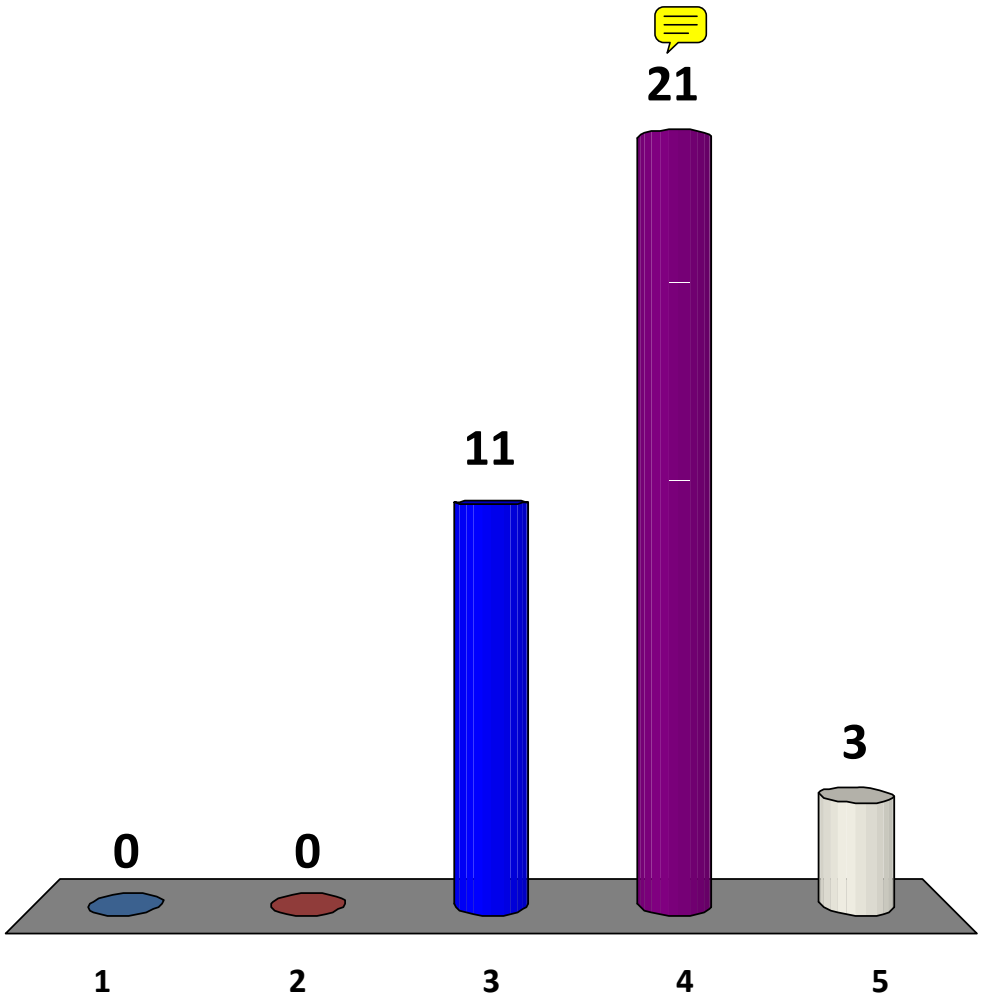
1. $O(N^2)$
2. O(N log N)
3. O(N)
4. O(log N)
5. O(1)



What is the best time complexity to search for an item in a sorted linked list of size N?

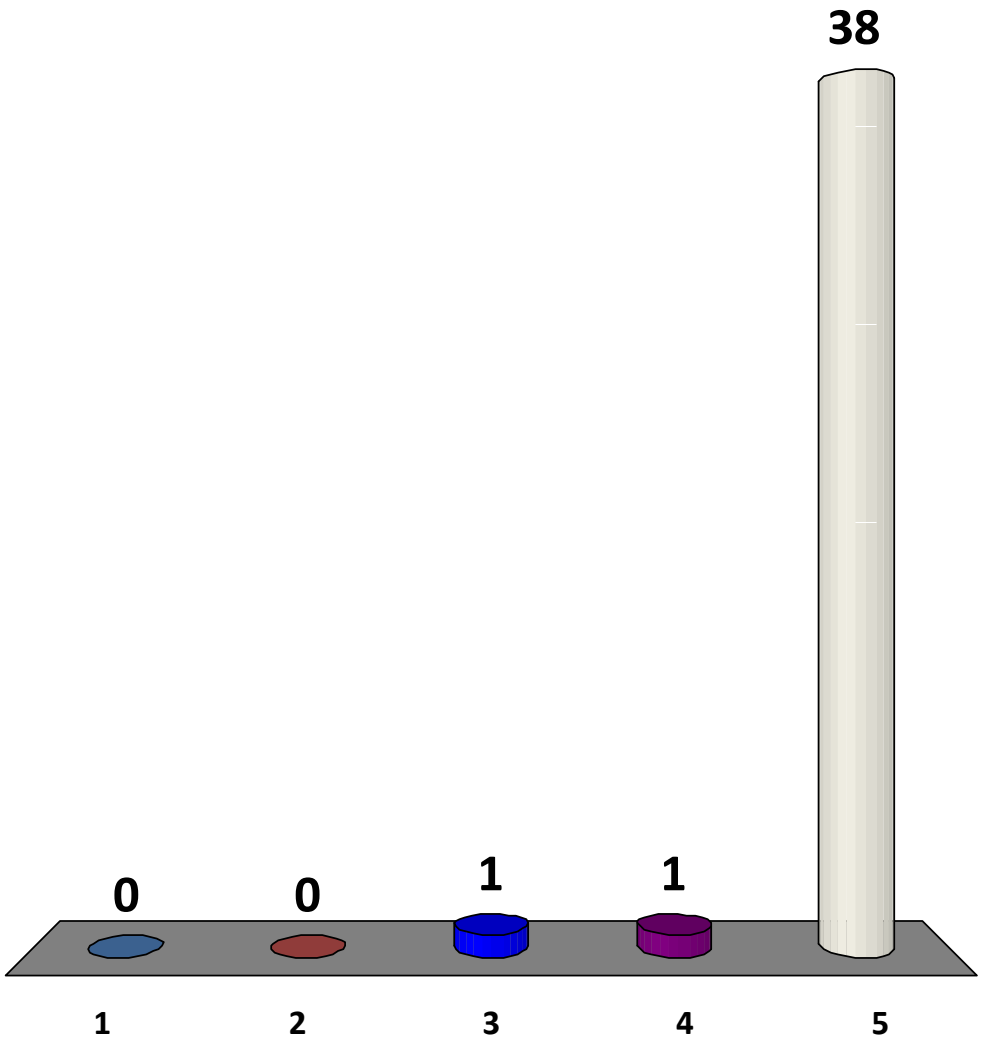
Hint: Binary Search?

1. $O(N^2)$
2. O(N log N)
3. O(N)
4. O(log N)
5. O(1)



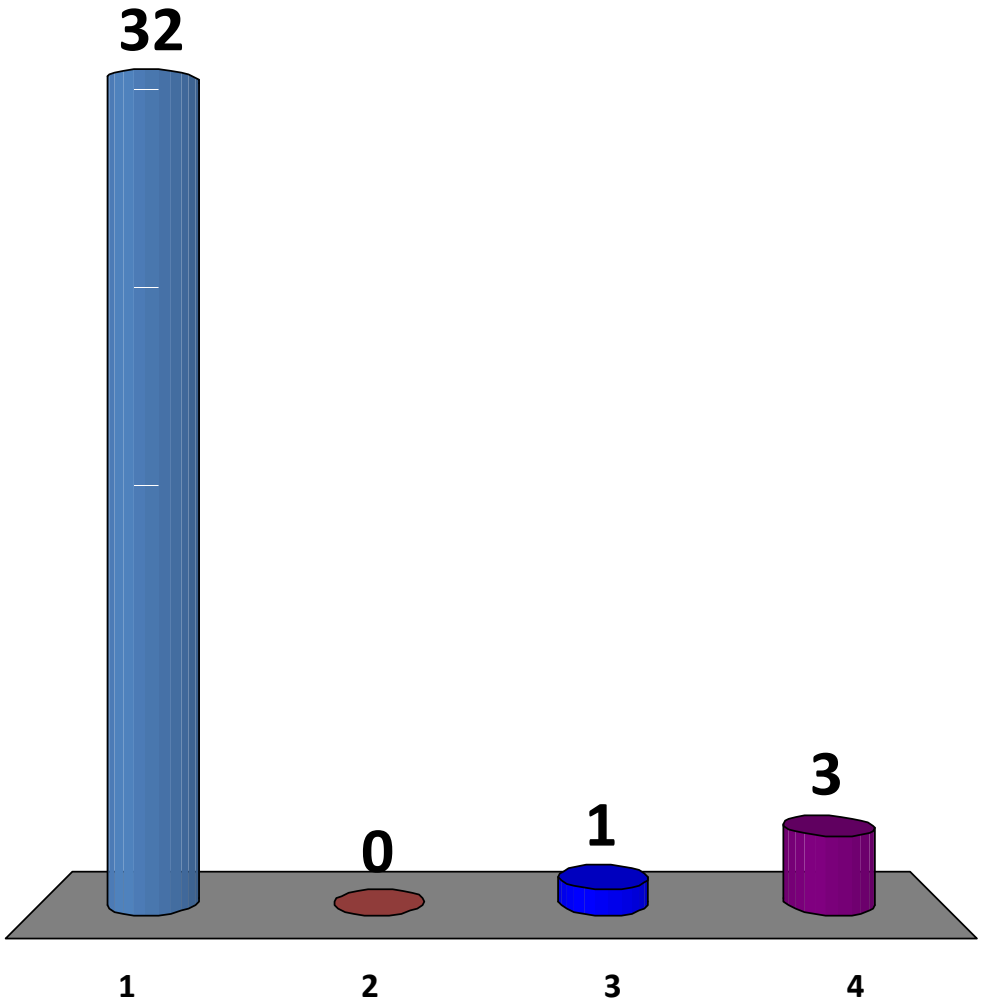
Four integers are inserted into a Stack one by one, then the top two are popped out, then the fifth integer is inserted into the same Stack. **Who is on top of the Stack now?**

1. The first integer
2. The second integer
3. The third integer
4. The fourth integer
5. The fifth integer



Three person “Steven”, “Grace”, “Felix” entered a queue, in that order. After waiting for a few minutes, the person in the front of the queue is called. **Who is he/she?**

1. Steven
2. Grace
3. Felix
4. Someone else



CS1020 – Sorting

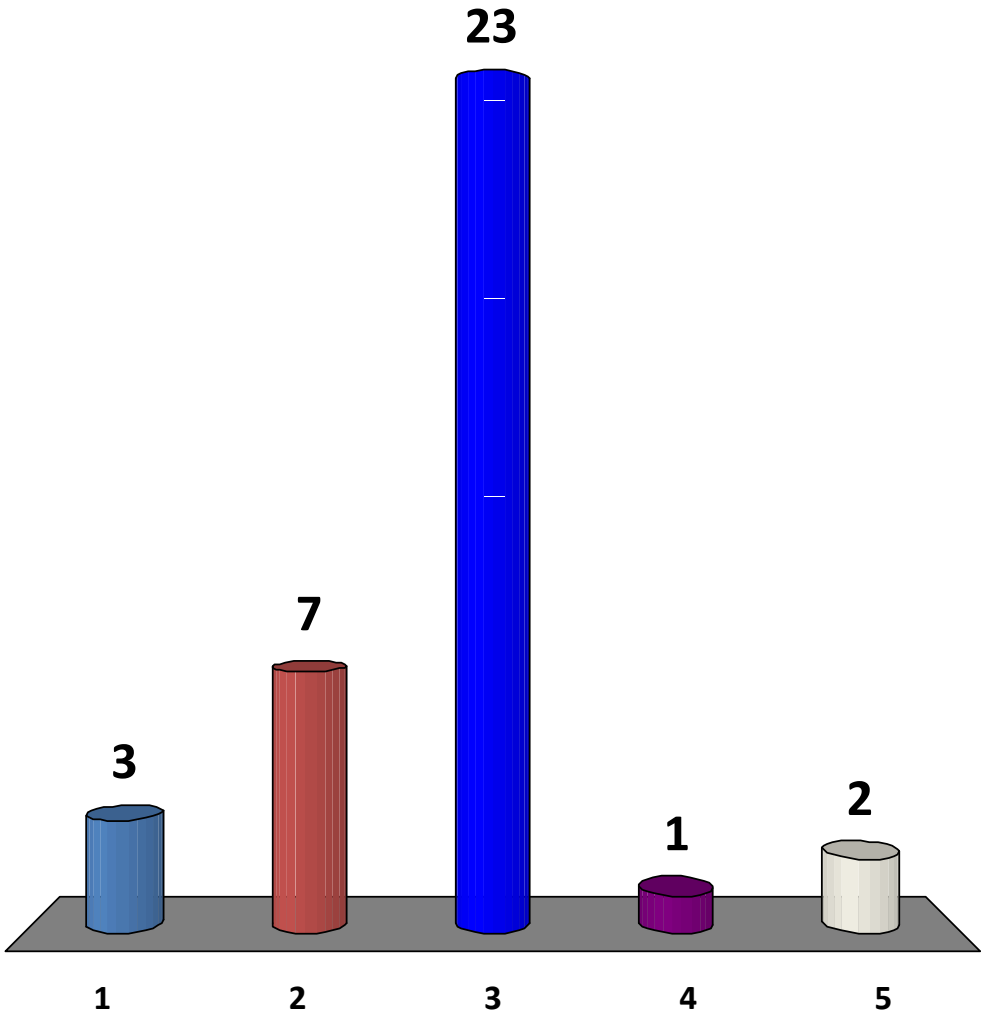
- What you learn in CS1020:
  - $O(N^2)$ Selection Sort, Bubble sort, Insertion sort
  - O(N log N) Merge sort
  - Expected O(N log N) Quick sort if the pivot is randomized
    - Can go to $O(N^2)$ otherwise (but this is what you learned in CS1020)
  - Visualization: 

    http://www.comp.nus.edu.sg/~stevenha/visualization/sorting.html
- In CS2010:
  - If not explicitly stated, you can use Java library functions, e.g. Collections.sort for all your sorting needs
  - We will learn more sorting algorithms: BST Sort, Heap Sort

What is the best sorting algorithm to sort this **almost sorted** sequence?

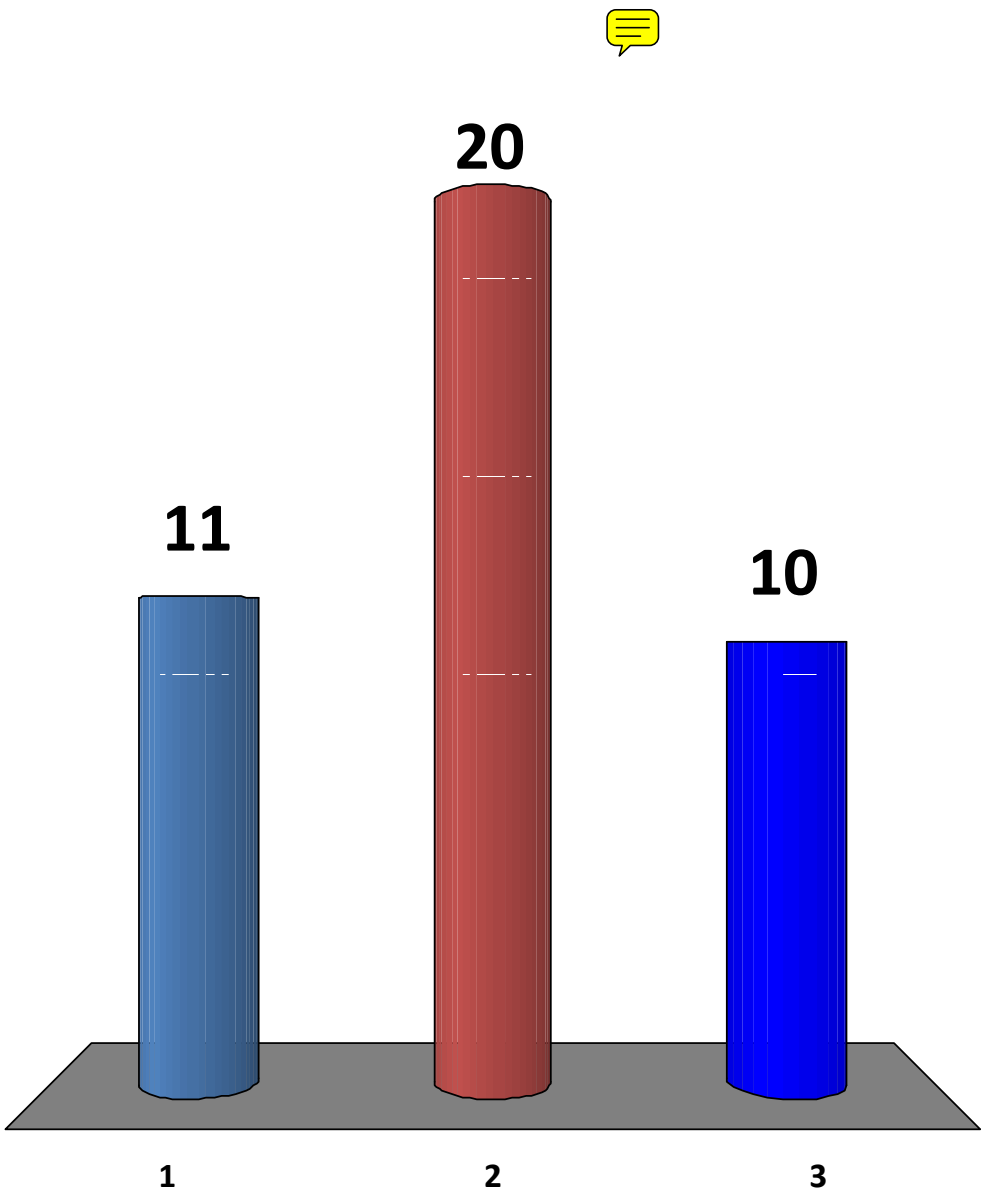
X = { 1, 1, 1, 1, 1, 3, 3, 4, 5, 6, 7, 2, 1M}

1. Selection Sort
2. Bubble Sort
3. Insertion Sort
4. Quick Sort
5. Merge Sort



I...

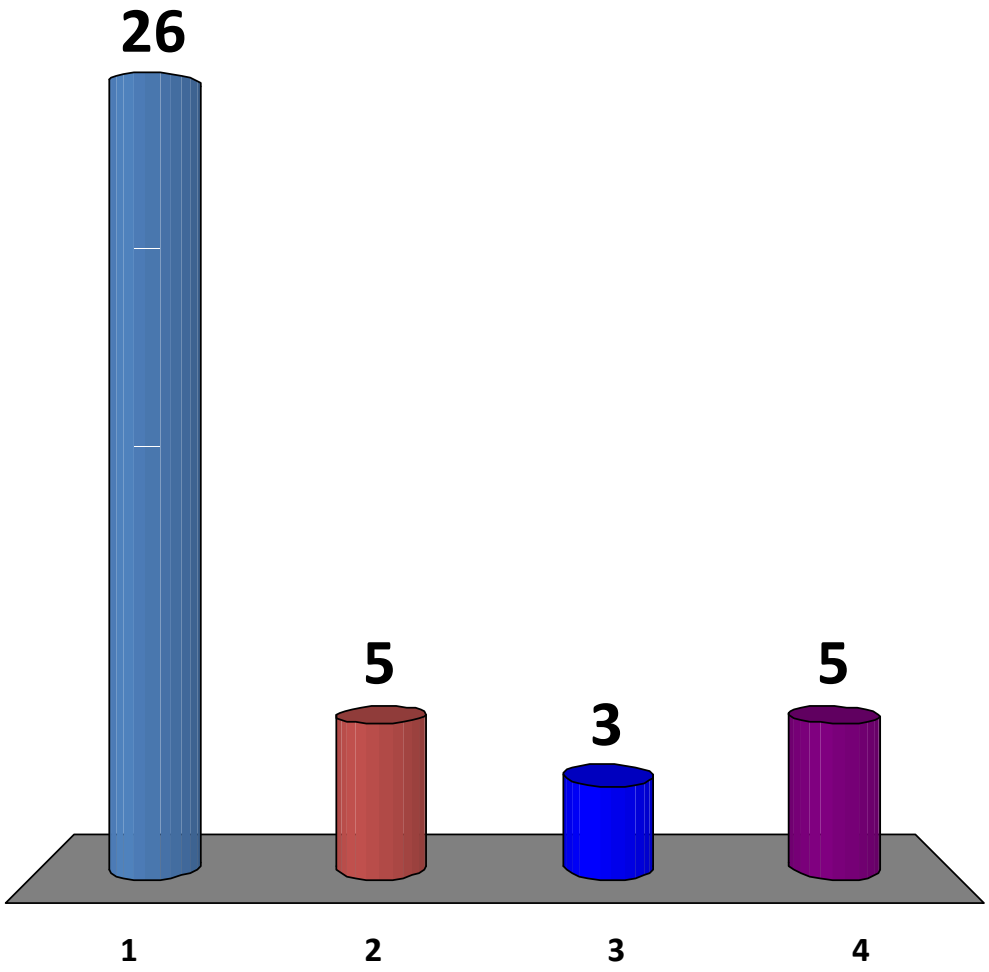
1. Have no problem with recursion examples shown in CS1020
2. Am lost with recursion
3. Am very lost with recursion ☹



Give me a pair of numbers x1 and x2 so that h(x1) = h(x2) for h(x) = (x * x) % 7

(you can select up to 4 options)

1. x1 = 71, x2 = 55
2. x1 = 77, x2 = 66
3. x1 = 7, x2 = 15
4. x2 = 9, x3 = 147

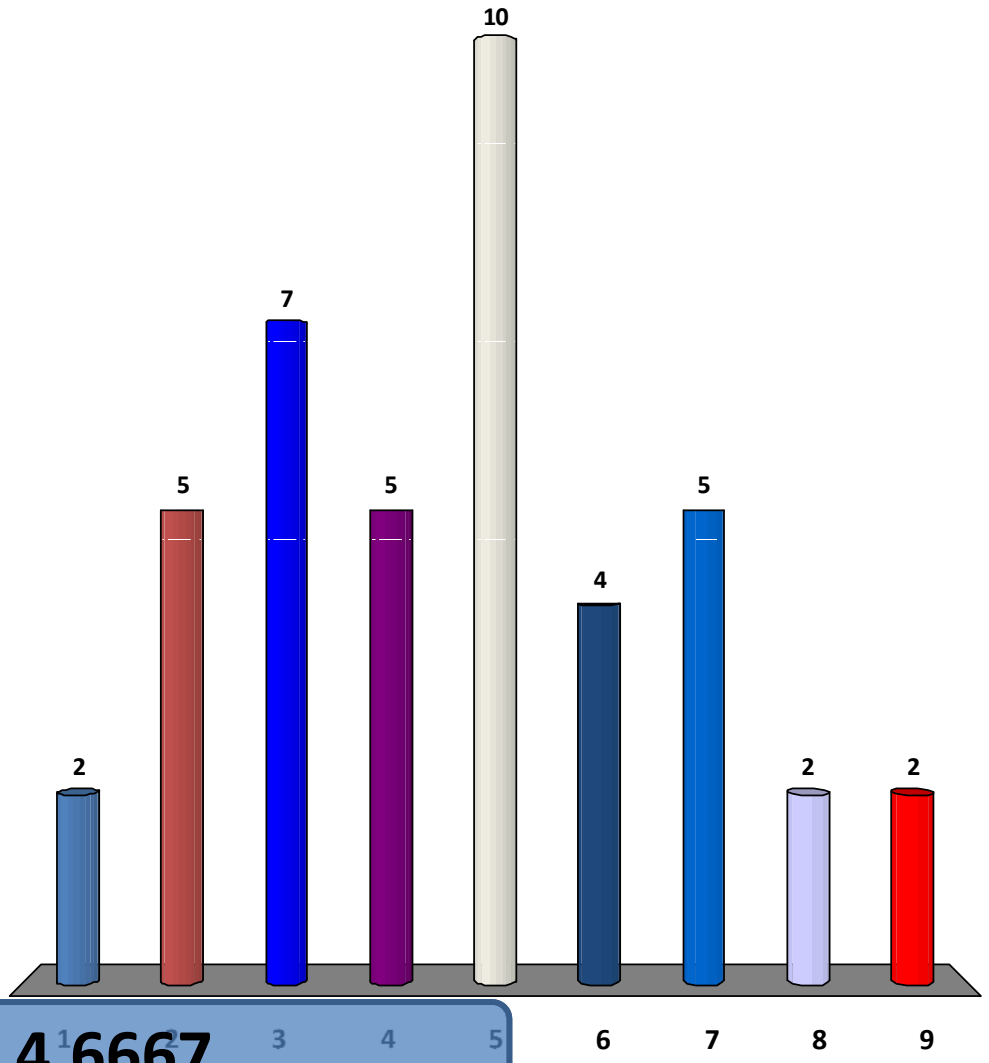


CS1231 – Discrete Structures

- Another relevant module for CS2010 is CS1231
- But CS1231 is not a pre-req of CS2010
- Relevant stuffs are:
  - Discrete structures: Graphs and Trees
    - Relevant Bryan Low's (old) CS1231 notes have just been uploaded
  - Proofs, we will see lots of them (simpler form)
- In CS2010, we will see all these discrete structures practically throughout the semester
  - That's it, lots of trees and graphs and proofs (simpler form)
- Let's see the profile of CS2010 students

My CS1231 (or MA1100) grade
(don't be shy, this is anonymous)

1. A+
2. A
3. A-
4. B+
5. B
6. B-
7. C+/C
8. D+/D/or F
9. Haven't take that mod!



Mean = 4.6667

Code For Science – Singapore

codeforscience.com/singapore

- First month of Sem I
- A chance for you to use expertise in your field of study to influence how researchers research
- Up to 5K SGD in prizes
- Special IDA prize to use data.gov.sg
- Work in teams of up to 4 people
- **New!** – You can earn SoC student leadership programme points by participating
- **New!** – Formally supported by CS 3108 and CS 4249 (CS 3213 in the works)

NUS ACM ICPC Teams 2012

http://algorithmics.comp.nus.edu.sg/wiki/training/icpc_workshop

NOC Presentation