CS2010 – Data Structures and Algorithms II

Lecture 01 – Introduction

chongket@comp.nus.edu.sg

NUS
National University
of Singapore

School *of* Computing

Welcome ☺

Introduction of Staffs

Lecturer:

Dr Chong Ket Fah
COM2-02-66

chongket@comp.nus.edu.sg

TAs (there are many):

See CS2010 IVLE

Outline

Some course administration:

- Assessments
- The IT tools used in this module:
  - VisuAlgo: Visualization/animation tool and Online Quiz tool,
  - Codecrunch: Platform for submission and grading of PSes
- The module itself
- Competitive Programming 3 (CP3) reference text

CS1020 Quick Review + Comparison with CS2010

- OOP, algo analysis, linear DS, recursion, sorting, hashing

Problem Solving Paradigms in the context of CS2010

- Complete Search, Divide & Conquer, Greedy, and Dynamic Programming

Assessments: Overview

Activities	Weightages
Tutorial+Lab attendance/participation	6 (3+3)%
Problem Sets	24%
2 Written Quizzes (open book+calculator)	20% (14+6)
2 Online Quizzes (open book+calculator)	10% (5+5)
Final Exam (open book+calculator)	40%

- Labs on Thursday
- Tutorials on Monday/Tuesday
- Open Book = allowed to bring in lectures notes, tutorials, quizzes, reference books or any piece of paper you want **but no internet!**

Assessment: Bi-Weekly PS (1)

There are **6*** Problem Sets (PSes) in CS2010

- Most of them are "related" ☺, and use real-life examples (some are a bit exaggerated)
- Each of the PSes have the same weightage (**4%** each)
- PS is based on a subtask system with each subtask have some marks
- Full marks for each subtask is given only if you pass **all test cases** for that subtask
  - For highest subtask attempted but failed, ¼ marks of that subtask will be given if solution is almost correct (this is a very strict criteria)

Assessment: Bi-Weekly PS (2)

Rules for Bi-Weekly PS:

- Collaboration at the **algorithmic level** is encouraged, but you have to write the solution (i.e. the Java code) **by yourself**!
  - Cannot discuss/show/copy each others' code
- Posting algorithm/data structure ideas to Facebook group is encouraged, but **never** send Java code to anyone **before deadline**, even if it is a "buggy" one!
  - You have to write and debug **your own** Java code!
  - Looking at your seniors' code is also considered as plagiarism
  - All submissions to CodeCrunch is **recorded**!
    Do NOT submit someone else's code using your account!
  - We will be using an *automatic plagiarism checker*, **be careful!**
  - **Note about stuffs like GitHub or ideone.com, make them private!**

Assessment: Bi-Weekly PS (3)

Rules for Bi-Weekly PS:

- **Offender caught cheating will be referred to the NUS Board of Discipline**

Assessment: Bi-Weekly PS (4)

PSes are the core of CS2010

- Most likely you will spend many hours (depending on your aim) discussing and implementing the solutions
  - It is designed as one “simple” problem with “subtasks” having *gradual* level of difficulty
    - The easier subtask just require CS1010-CS1020 knowledge
    - Most subtasks definitely require CS2010 knowledge
    - The last subtask may require more creative usage of CS2010 knowledge or higher level knowledge
- The ideas (not the Java code) that can solve Subtask 1-2 of each PS will be discussed during the tutorial sessions ☺
  - So, you can score up to 25-50 (out of 100) marks by understanding what is discussed in the tutorial and then implementing it

Assessment: Bi-Weekly PS (5)

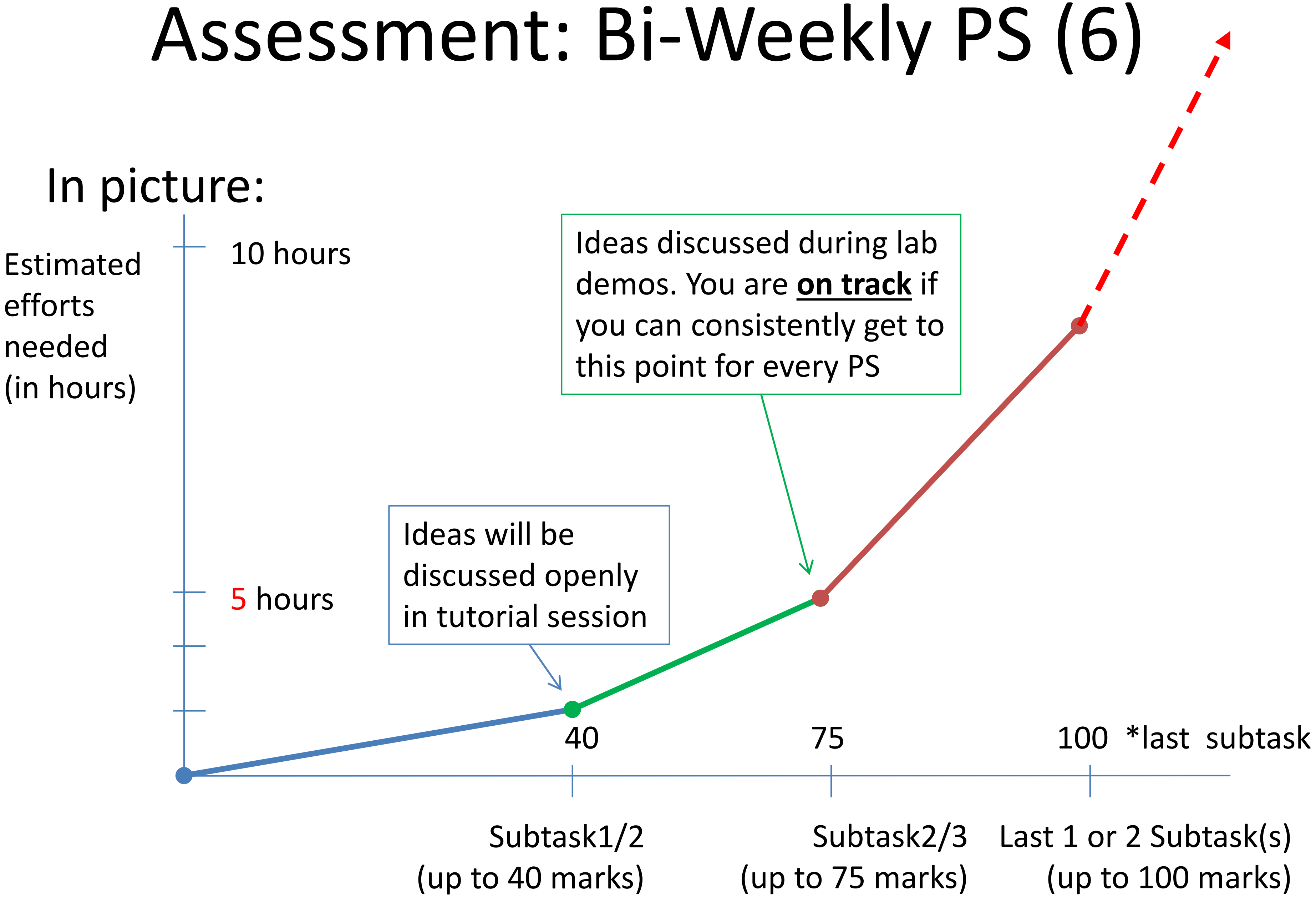
PSes are the core of CS2010 (continued)

- The implementation of the required technique (but not the actual solution) that can be used to solve parts of the harder subtasks will be discussed during the lab demos
  - You can score up to 50-75 (out of 100) marks by understanding what your Lab TA is trying to tell you during his/her lab demo
- The last subtask of each PS is designed for those who are aiming to get A/A+ in this module… they are either ***difficult*** or ***tedious***
  - If this is your aim, you may end up spending *hours* to solve this
  - If you aim for A/A+ you will need to do this, otherwise don’t spend too much time on it, but I will still recommend everyone to try it

☺

Assessment: Bi-Weekly PS (6)

In picture:



Ideas will be discussed openly in tutorial session

Ideas discussed during lab demos. You are **on track** if you can consistently get to this point for every PS

Assessment: Written Quizzes 20%

2 Written Quizzes

- Quiz 1 **14%** - 6th week (Friday 22nd Sep, 7pm – 8:30pm),
  just before recess week **please don't go on holiday on Friday!**
- Quiz 2 **6%** - 12th week (Wednesday 8th Nov during lecture),
  shorter one to prepare you for final exam

- PSes deal with more with your ability to come up with and implement the algorithm
- Quizzes/Final assessment deal more with your ability to model the problem correctly and come up with the algorithm to solve it (only need pseudo-code)

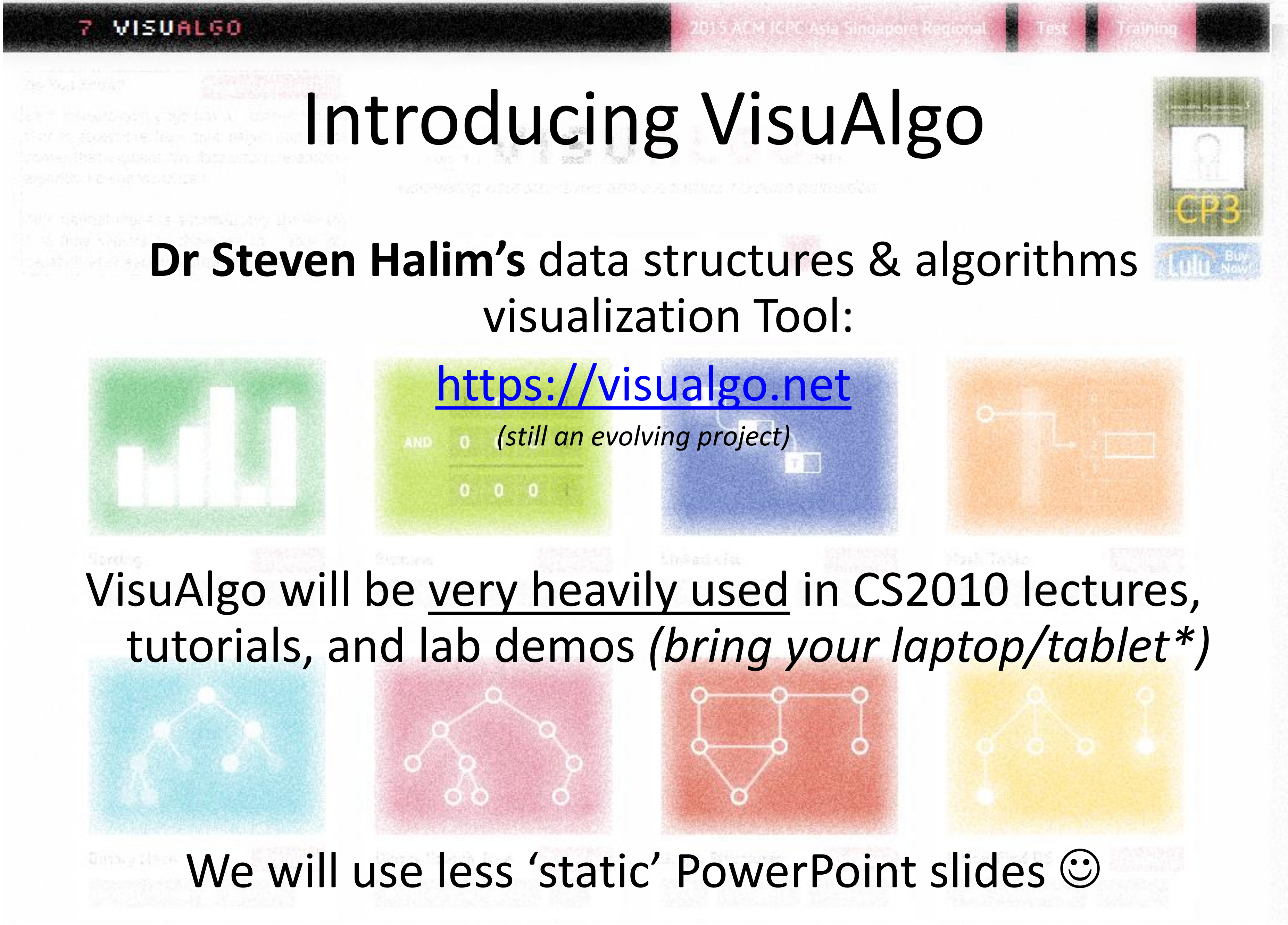
Assessment: Online Quizzes **10%**

2 online quizzes

- Quiz 1 **5%** - 6th week (21st Sep) during lab
- Quiz 2 **5%** - 11th week (2nd Nov) during lab

- Test basic to intermediate knowledge on the algorithms/data structures learned

Introducing VisuAlgo



Dr Steven Halim's data structures & algorithms visualization Tool:

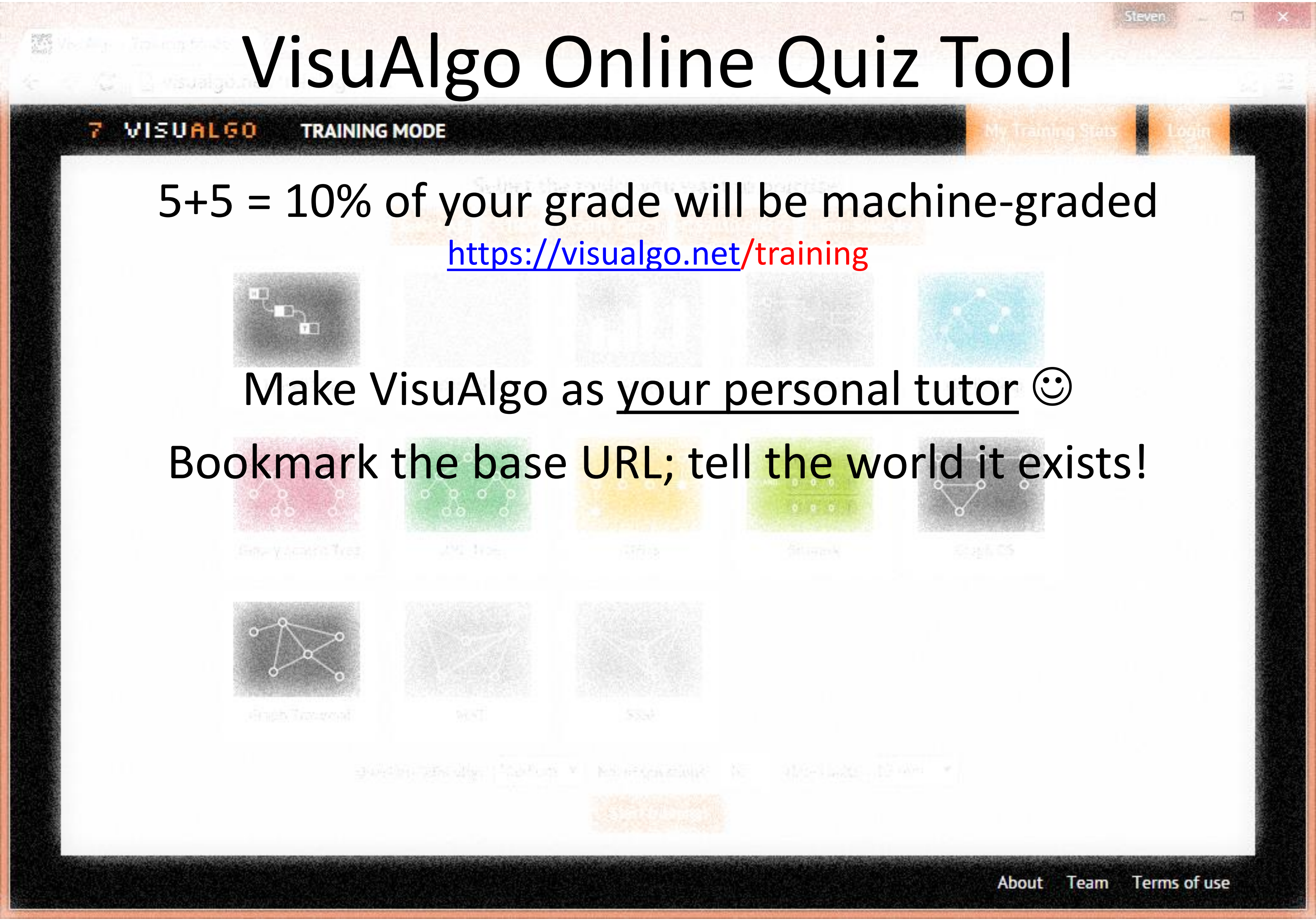
https://visualgo.net

(still an evolving project)

VisuAlgo will be very heavily used in CS2010 lectures, tutorials, and lab demos *(bring your laptop/tablet*)*

We will use less 'static' PowerPoint slides ☺

VisuAlgo Online Quiz Tool



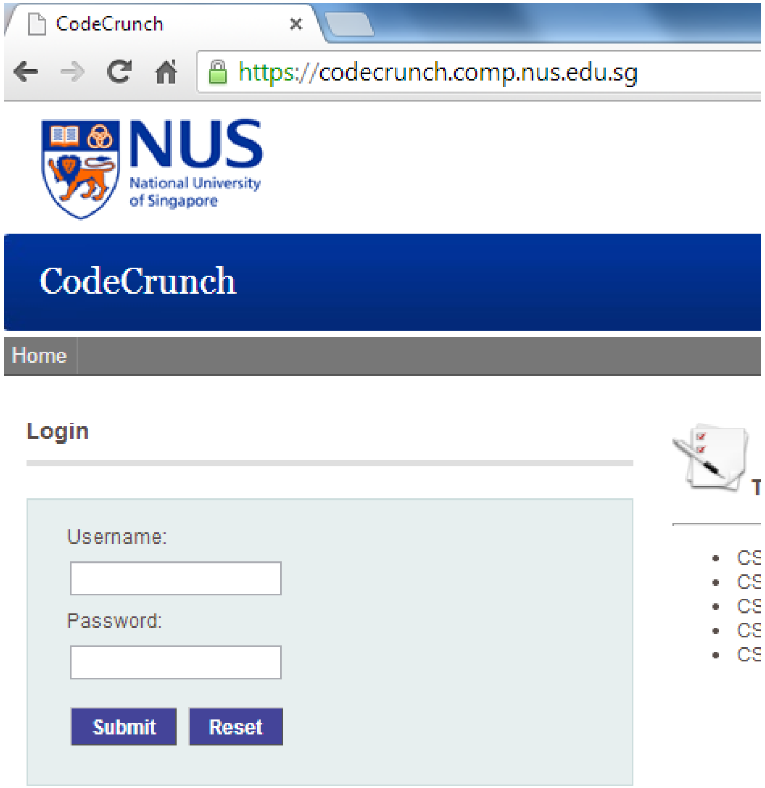
5+5 = 10% of your grade will be machine-graded

https://visualgo.net/training

Make VisuAlgo as your personal tutor ☺

Bookmark the base URL; tell the world it exists!

CodeCrunch - https://codecrunch.comp.nus.edu.sg/



- Guide on using CodeCrunch is included in the lecture notes package

CodeCrunch - https://codecrunch.comp.nus.edu.sg/

- **PS0 is up on CodeCrunch for you to try.**
- Unlike for CS1010/CS1020 there will not be output shown when there is an error between your result and the expected result !
  - Have to test your program first using test cases you construct
  - Learn to create good test cases -> **very important skill** !
- Other then compilation error, what you will see is (not exhaustive)
  - Time Limit Exceeded
  - Memory Limit Exceeded
  - Incorrect Output
  - Runtime exception/error

THE MODULE…

How to get B-/B grade in CS2010

Very simple

Just do the minimum requirements

- Get slightly better than passing for your written quizzes/exam
- get ~75% for all PSes
- get ~8 out of 10 for online quizzes
- **Attend all your labs+tutorials**

How to get B+/A- grade in CS2010

Do all those required for B-/B grade

Improve your mathematics proficiency

(One indicator is your performance on CS1231 or MA1100)

<u>Occasionally</u> complete the harder subtasks of the PSes

+

<u>Occasionally</u> be able to solve the harder questions in the written quiz and final exam

How to get A/A+ grade in CS2010

Summary: Do everything that is graded with near perfect score…

A/A+ students in CS2010 are invited to take Dr Steven Halim’s CS3233 course in Sem2 AY 2017/2018

Discussion Forum on Facebook not IVLE!

We have a **Facebook** page!

https://www.facebook.com/groups/241724769269875

This is where discussions will be held instead of IVLE forum

Join now (if you don’t have FB account, register)!**

What’s IVLE for then?

- Publishing Lecture Notes/Tutorials
- Annoucements
- Gradebook

CP3 Book Sales

Not compulsory (actually a CS3233 text book)

Written by Dr Steven Halim and contains about ~65% of his algorithmic knowledge so far

- **Might be** useful to tackle the last subtask of each PS and to answer some tricky questions during Quiz1/2/Final (maximum 20-25% per test)

Will sell the book **today** (16th August) from 2pm to 7pm

- Come to my office at COM2-02-66 to buy it
- 22 SGD/copy
- will have 50 copies for sale

Alternative option: Borrow @ CL

- RBR (Reserved Books/Readings) QA76.6 Hal 2013

5 minutes break, and then…

CS1020 REVIEW WITH HELP OF VISUALGO

CS1020 – OOP

Object Oriented Programming (OOP) in Java

```
class BankAccount {
  private int Balance;
  public BankAccount();
  public void Deposit(int Amount);
  public void Withdraw(int Amount);
  public int CheckBalance();
}
```

We will use Java classes/OOP principles in our CS2010 PSes

- e.g. We use "IntegerScanner", "IntegerPair", and many more in CS2010

CS1020 – Basic Algorithm Analysis

Big O notation, the O(g(n)) stuffs, e.g.

```
sum = 0;
for (int i = 0; i < n; i++)
  sum += A[i];
// is an O(n) algorithm
```

In CS2010, we will

- Extensively use this kind of algorithm analysis
- Learn a few more advanced algorithm analysis skills

CS1020 – Linear Data Structures (1)

Data Structure is a way to store, organize and answer queries about data

- We will frequently abbreviate it as **DS**

A good DS is needed to support *efficient*:

- **Insertions**: add a new item into the DS
- **Searches**: is item X inside the DS or not?
- **Deletions**: remove a certain item out from the DS
- **Queries**: how many items is the DS?, what is the min item in the DS?
- **Updates**: combination of (or a more efficient form than) "delete the old item" and "insert the new item"

Different situations may require different DS

CS1020 – Linear Data Structures (2)

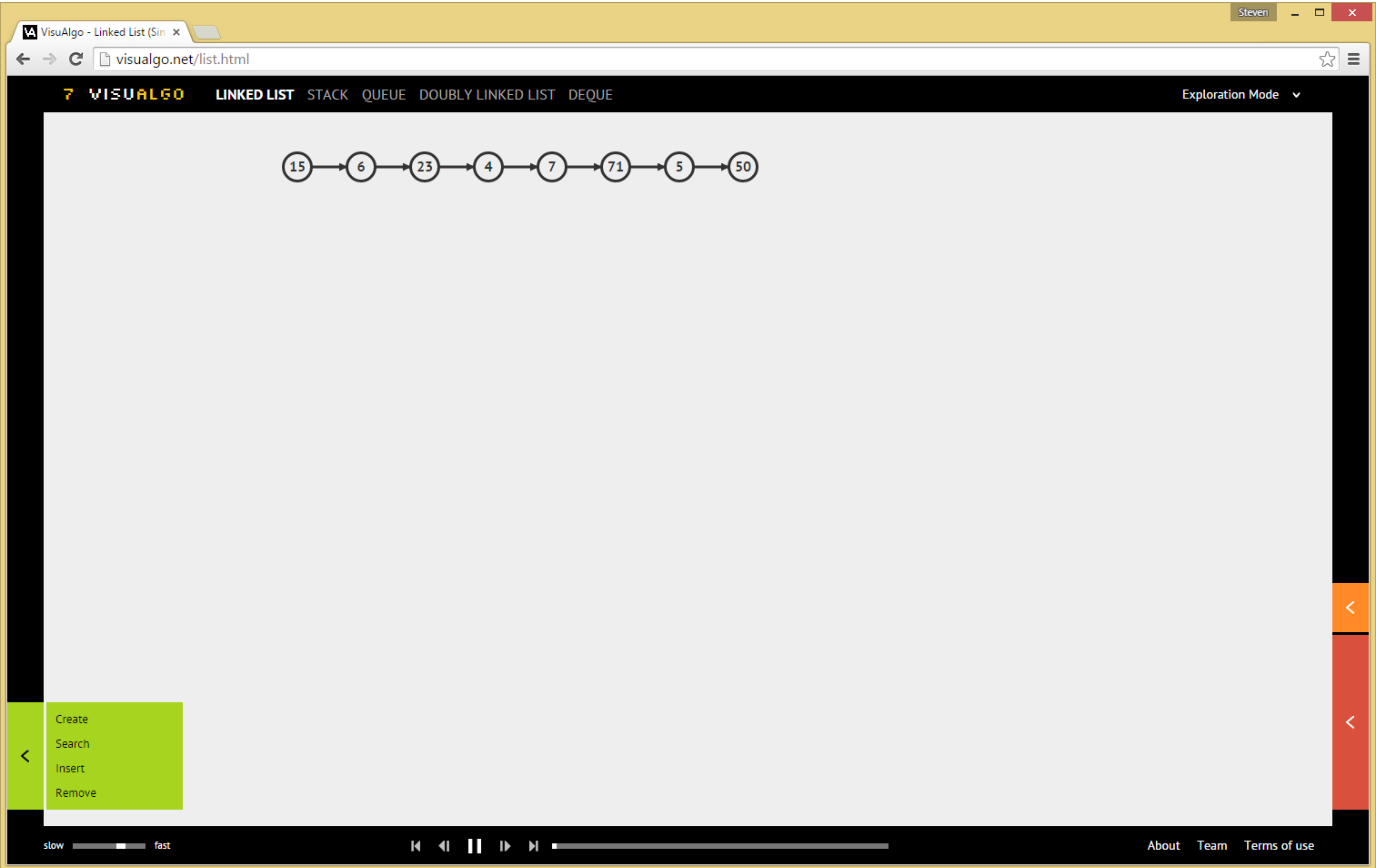
Linear DSes that you learned in CS1020:

- Items listed in left-to-right (or top-to-bottom) order
  - **Array** (fixed size)/**Vector** (resizeable)
  - **Linked List**
  - **Stack**: Last In First Out (LIFO)
  - **Queue**: First In First Out (FIFO)

In CS2010 you will learn non-linear DSes:

- (Binary) Heap, Binary Search Tree, Union Find DS, Graph

Linked List Visualization



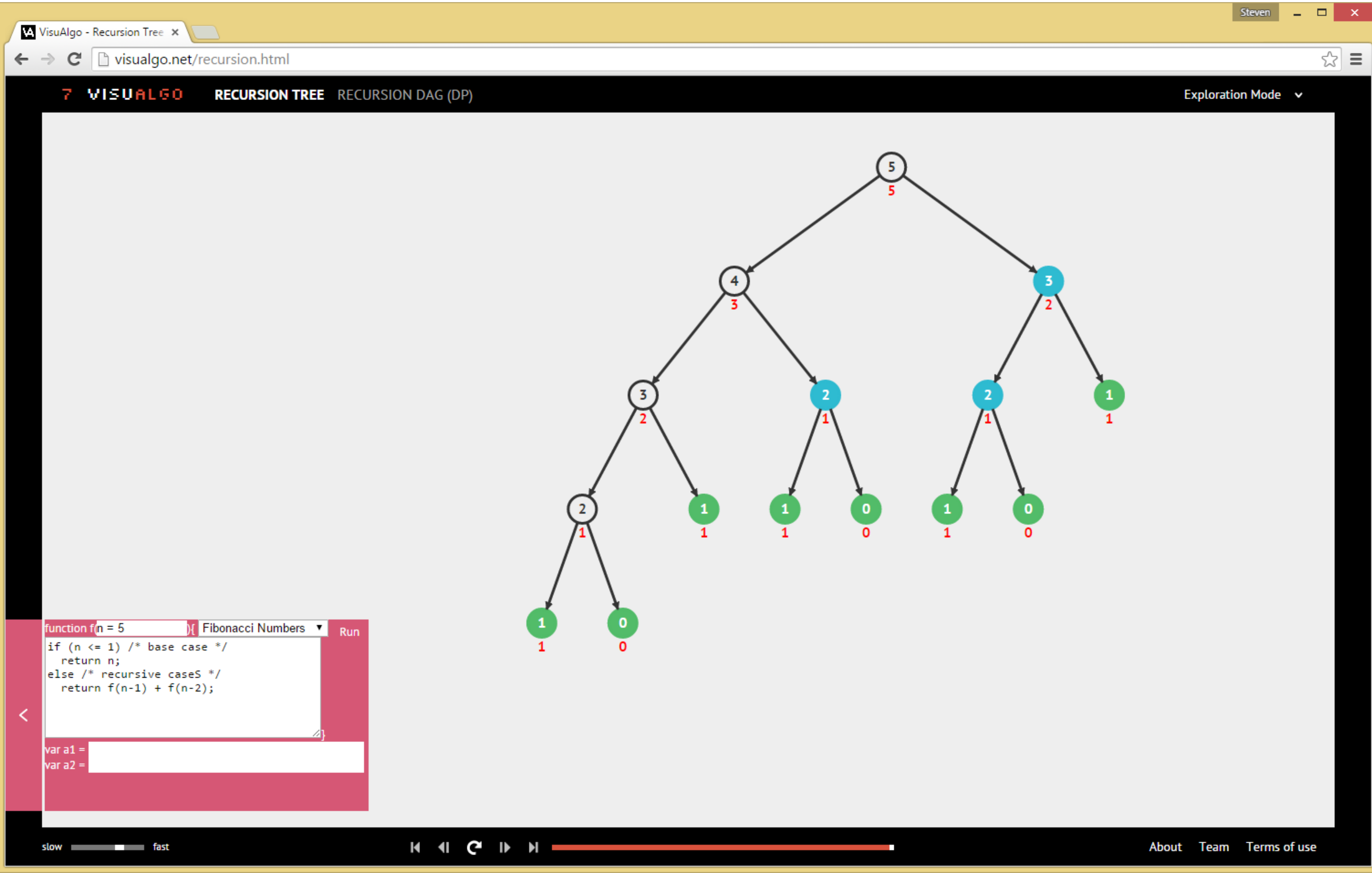
CS1020 – Recursion

In CS1020, you may have learned these examples:

- “Countdown”
- Factorial
- Printing a linked list in reverse order
- Towers of Hanoi
- N choose K
- String permutations
- Recursive binary search
- Fibonacci

In CS2010, we will see *much more* recursion

Recursion Tree/DAG Visualization



function f(n = 5){ Fibonacci Numbers

```
if (n <= 1) /* base case */
  return n;
else /* recursive caseS */
  return f(n-1) + f(n-2);
```

}

var a1 =
var a2 =

CS1020 – Sorting

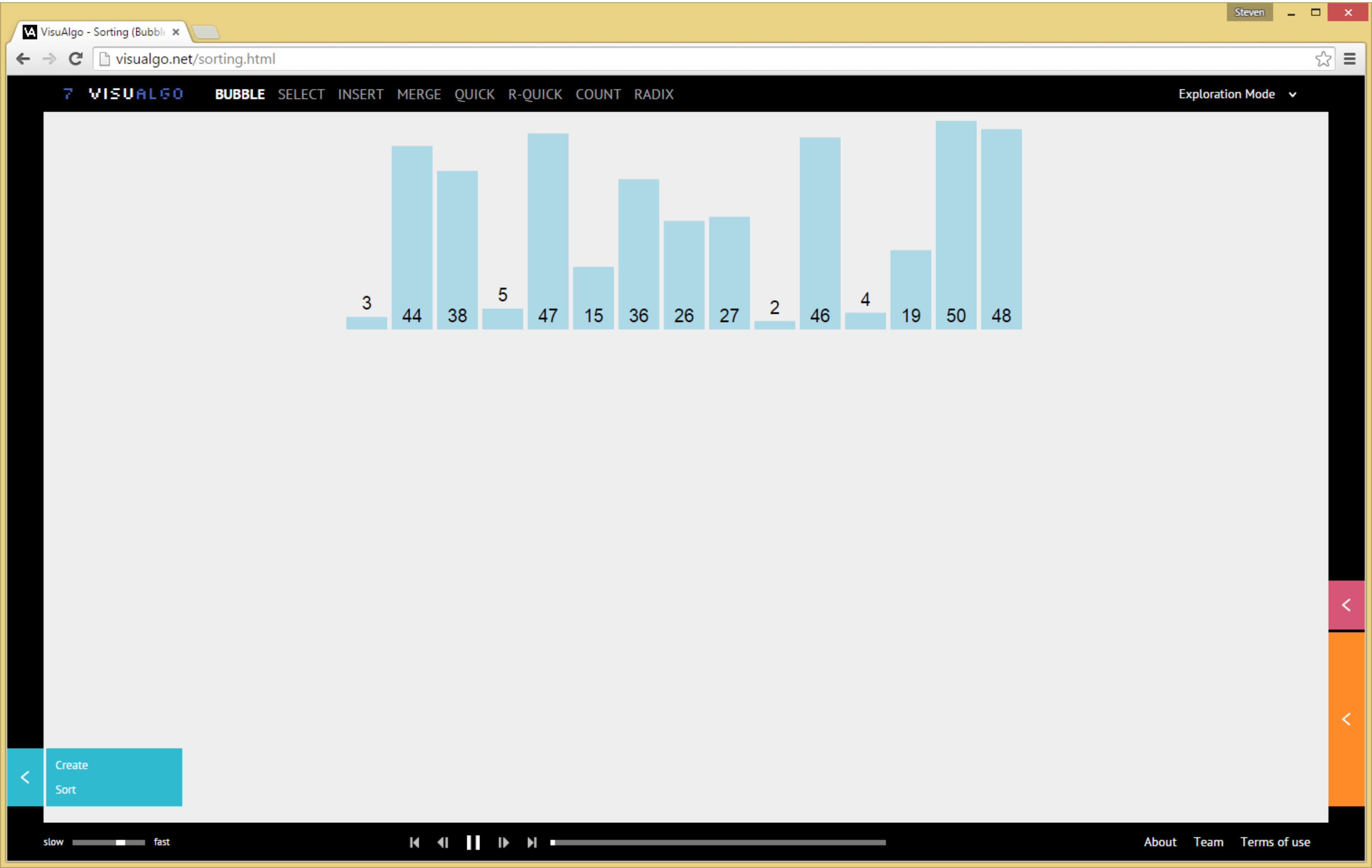
What you learn in CS1020:

- $O(N^2)$ 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 run in $O(N^2)$ otherwise (this is what you learned in CS1020)
  - A stronger analysis of Randomized Quicksort later in CS3230

In CS2010:

- If not explicitly stated, you can use Java API,
  e.g. Collections.sort for all your sorting needs
- We will learn more sorting algorithms: Heap Sort, BST Sort

Sorting Visualization



CS1020 – Hashing

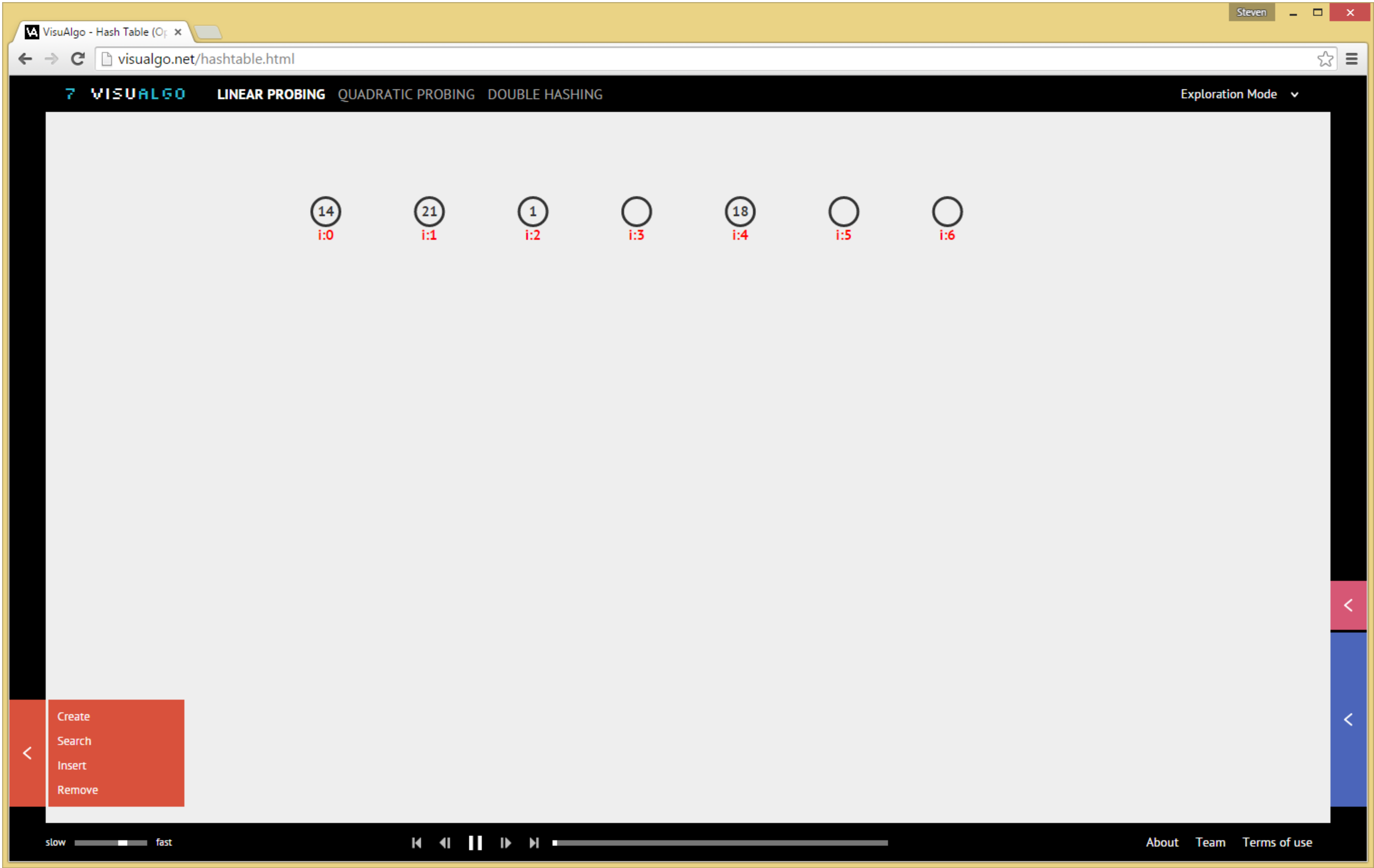
Concepts that you learn:

- Direct Addressing Table
- Creating good Hash Function
- Handling collisions: Birthday paradox
  - Separate chaining
  - Linear probing, modified linear probing, quadratic probing, double hashing

We won’t relearn hashing again in CS2010

- But you may have to contrast and compare it with BST later

Hash Table


VisuAlgo - Hash Table (O
visualgo.net/hashtable.html
Steven
7 VISUALGO
LINEAR PROBING
QUADRATIC PROBING
DOUBLE HASHING
Exploration Mode
14
i:0
21
i:1
1
i:2
i:3
18
i:4
i:5
i:6
Create
Search
Insert
Remove
slow
fast
About
Team
Terms of use

Let’s Review CS1020!

- VisuAlgo Online training
  - CS1020 Review Questions

Source:

-. Competitive Programming 3, Chapter 3 (overview)

-. Introduction to Algorithms, 2nd ed, Chapter 7 and 15-16

This is what we will learn in CS2010 ☺

PROBLEM SOLVING PARADIGMS

Complete Search

Given an integer array A = {10, 7, 3, 5, 8, 2, 9}, n = 7

Find the largest and the smallest element of A!

Divide and Conquer

Given an integer array A = {10, 7, 3, 5, 8, 2, 9, …},

but now n = 100000 items

What is the 12345th smallest item in A?

Is the previous Complete Search algorithm suitable?

Greedy

Given an integer array A = {10, 7, 3, 5, 8, 2, 9, …}
  n is still 100000 items

Find the largest gap *g* such that $x, y \in A$ and $g = |x - y|$

Dynamic Programming

Given an integer array A = {10, 7, 3, 5, 8, 2, 9, …} but now n = 1000 items

What is the **longest subsequence** of A that if viewed from left to right is always non decreasing?

- {3, 5, 8} is a subsequence, and non decreasing
- {3, 5, 8, 2} is also a subsequence, but 8 → 2 is decreasing
- {3, 5, 8, 9} is the longest so far (ignoring the '…')

In the Context of CS2010 (1)

Lecture 02

- A **Divide and Conquer** principle in Data Structure
- Heap DS and operations on it

Lecture 03-04

- Another **Divide and Conquer** principle in Data Structure
- Binary Search Tree (BST) and operations on it
- Balanced BST: Adelson-Velskii Landis (AVL) Tree

In the Context of CS2010 (2)

Lecture 05

- A few more Data Structures
  - Union-Find Data Structure
  - Basic Graph Data Structure
- **(Quiz 1 is up to here)**

Lecture 06

- Graph Traversal
  - Breadth-First and Depth-First Search
  - Their applications (usually classified as Complete Search)

In the Context of CS2010 (3)

Lecture 07

- Minimum Spanning Tree (MST)
  - Prim's and Kruskal's are both Greedy algorithms

Lecture 08-09:

- Single-Source Shortest Paths (SSSP)
  - Bellman Ford's, Dynamic Programming
  - Dijkstra's, Greedy algorithm
- **(Quiz 2 is up to here)**

In the Context of CS2010 (4)

Lecture 10-11-12

- Algorithms on DAG
- Classical DP
- All-Pairs Shortest Paths
  - All use Dynamic Programming
- **(Final exam is up to here)**

Lecture 13

- Review lecture + Special Topic? (not tested)

That’s all for today

We will gear up for the first main topic of CS2010

- ADT Priority Queue and Binary Heap Data Structure

To do list at home:

- Review CS1020 material about Array, Linked List, Queue and Recursion