CS1010 Programming Methodology

A beginning in problem solving in Computer Science

Aaron Tan

http://www.comp.nus.edu.sg/~cs1010/

24 July 2017

NUS National University of Singapore | School of Computing

Announcements

This document is available on the CS1010 website

http://www.comp.nus.edu.sg/~cs1010/4_misc/freshmen.html

Announcements

- Choosing CS1101S over CS1010
  - URL: https://register.comp.nus.edu.sg/UGOffice4
  - Deadline: **27 July 2017, Thursday, 6pm**
  - Default: CS1010 (No action needed if you decide to take CS1010)
  - No bidding queue in CORS for both CS1010 and CS1101S
  - Students from IS Dept are precluded from taking CS1101S

DDP students in CS and Maths/Applied Maths will be pre-allocated CS1101S.

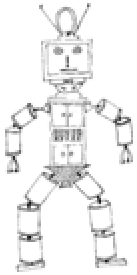
For more details, please contact SoC Undergraduate Office @ COM1, Level 2, Room 19

Computer Science Curricula 2013

- Ironman draft identifies 18 knowledge areas
  (http://ai.stanford.edu/users/sahami/CS2013/ironman-draft/cs2013-ironman-v1.0.pdf)

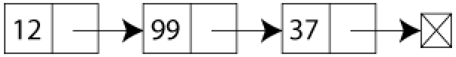
- AL - Algorithms and Complexity
- AR - Architecture and Organization
- CN - Computational Science
- DS - Discrete Structures
- GV - Graphics and Visual Computing
- HCI - Human-Computer Interaction
- IAS – Security and Info Assurance
- IM - Information Management
- IS - Intelligent Systems

- NC – Networking and Communication
- OS - Operating Systems
- PBD – Platform-based Development
- PD – Parallel and Distributed Computing
- PL - Programming Languages
- SDF – S/W Dev. Fundamentals
- SE - Software Engineering
- SF – Systems Fundamentals
- SP - Social and Professional Issues



$O(n^2)$

P = NP ?



CS1010

Introduces the *fundamental concepts* of *problem solving by computing and programming* using an imperative programming language.

Outcomes

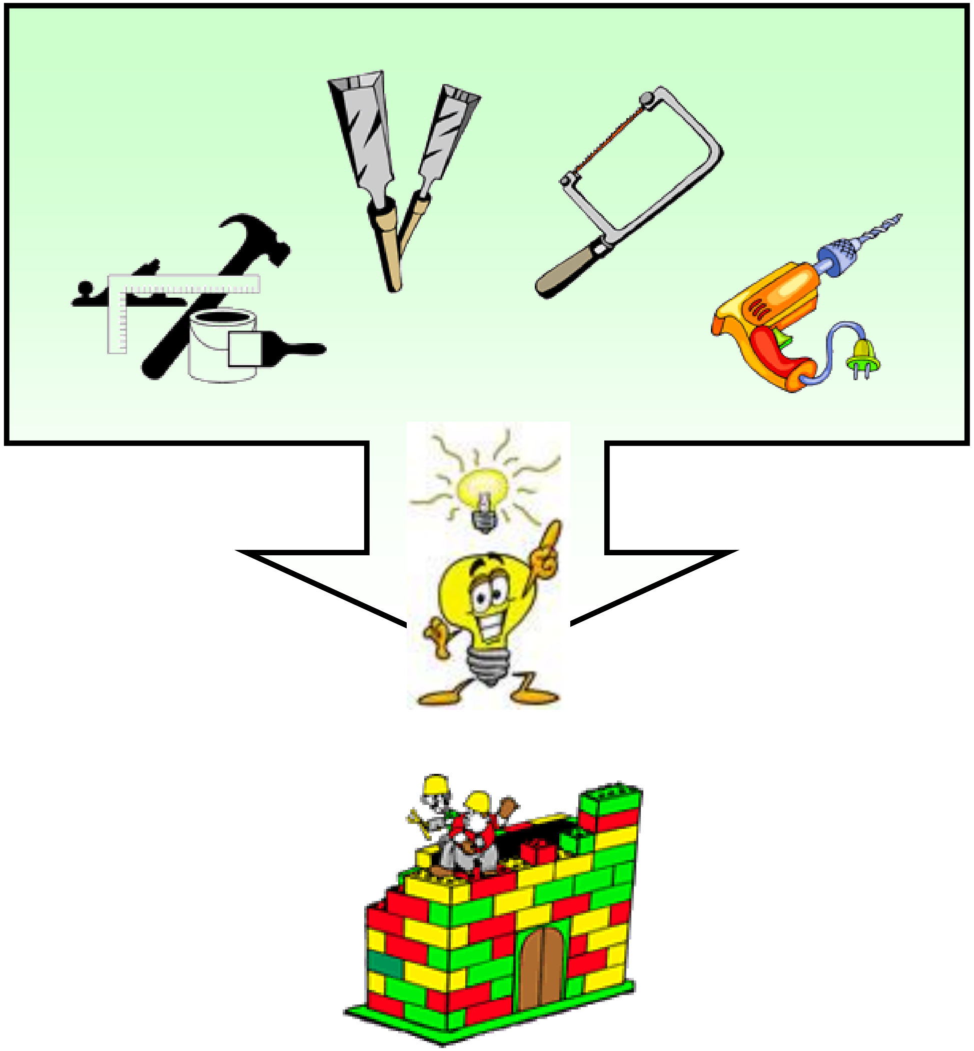
C as a tool

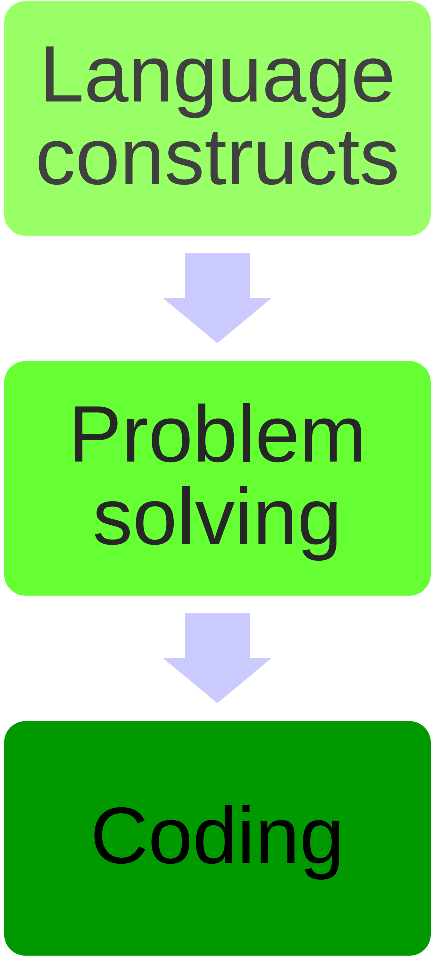
Solve simple algorithmic problems

Write good small programs

Not just about C

Programming





Language constructs

Problem solving

Coding

A C Program (welcome.c)

library

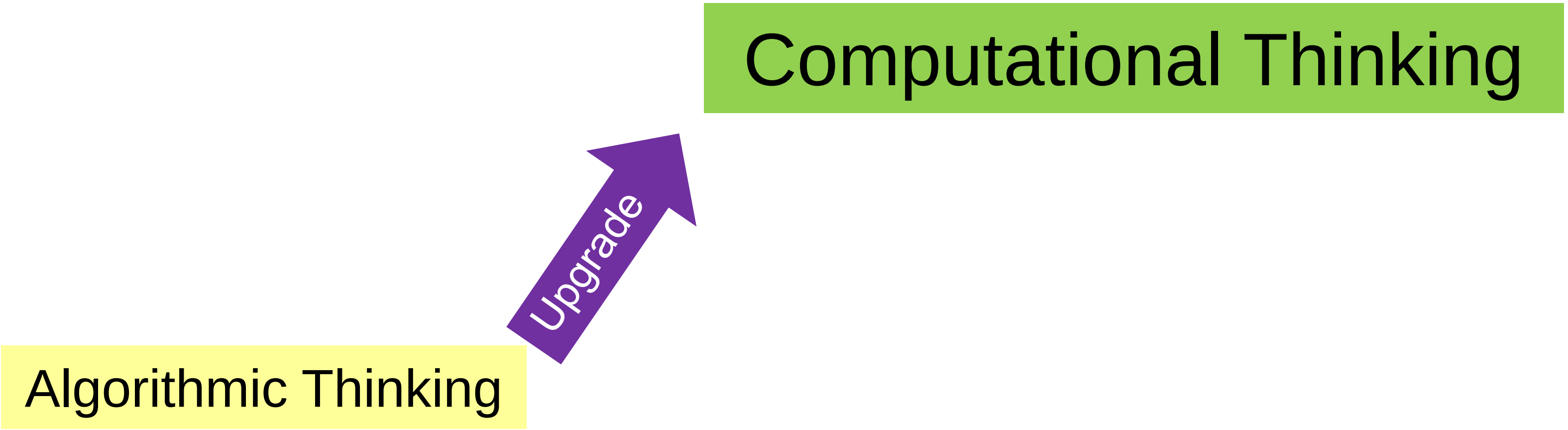
```c
// Author: Aaron Tan
// Purpose: Ask for user’s name and display a welcome message.

#include <stdio.h>

int main(void) {
      char name[20];
      printf("What is your name? ");
      scanf("%s", name);
      printf("Hi %s.\n", name);
      printf("Welcome to CS1010!\n");
      return 0;
}
```

```
What is your name? Aaron
Hi Aaron.
Welcome to CS1010!
```

Input

Outputs

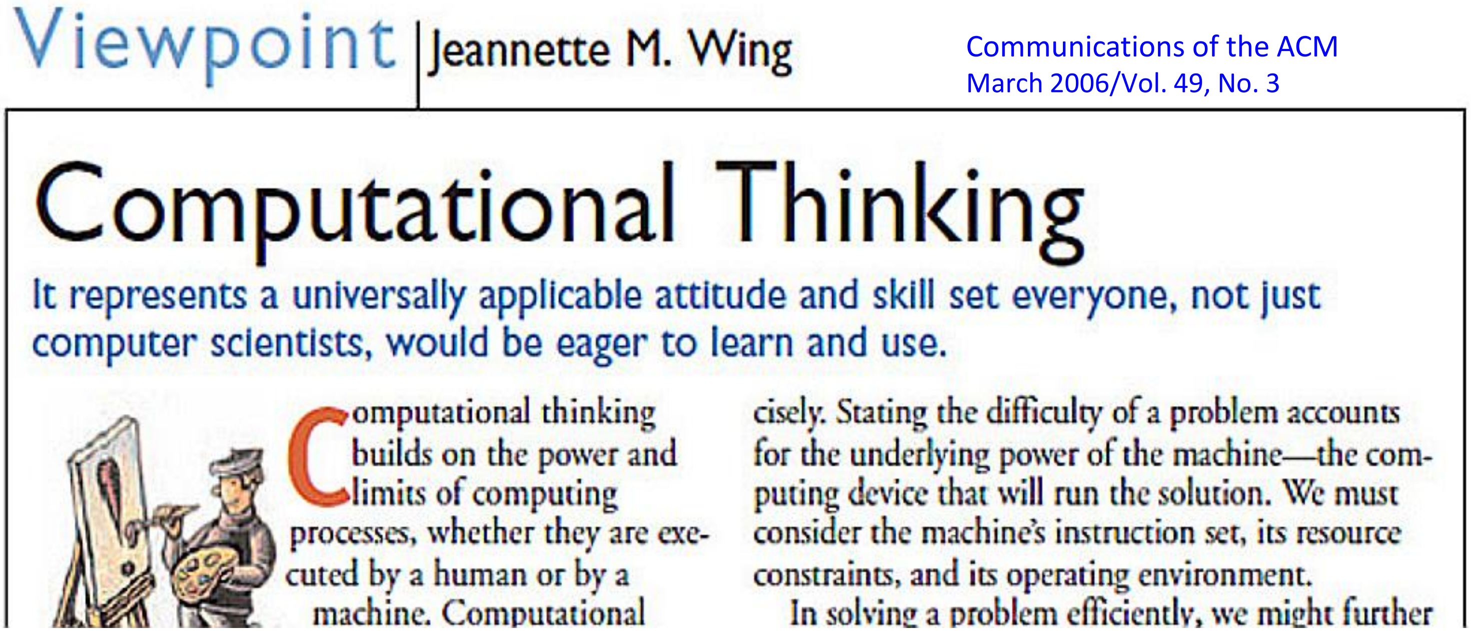
Problem Solving Skills



Computational Thinking

Upgrade

Algorithmic Thinking



Viewpoint | Jeannette M. Wing

Computational Thinking

It represents a universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use.

Computational thinking builds on the power and limits of computing processes, whether they are executed by a human or by a machine. Computational

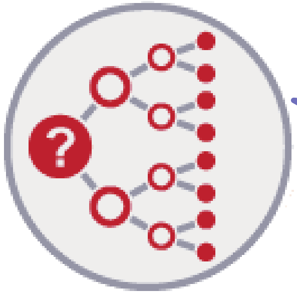
cisely. Stating the difficulty of a problem accounts for the underlying power of the machine—the computing device that will run the solution. We must consider the machine's instruction set, its resource constraints, and its operating environment.

In solving a problem efficiently, we might further

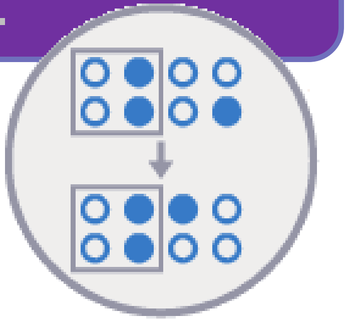
Communications of the ACM
March 2006/Vol. 49, No. 3

Computational Thinking

Decomposition



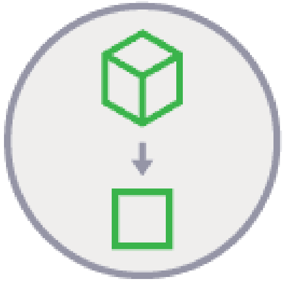
Breaking the problem into smaller, more manageable parts.



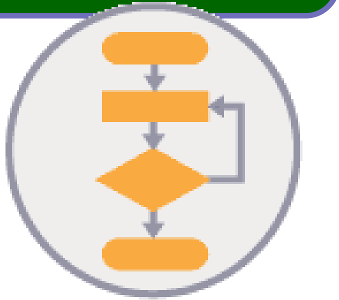
Recognising which parts are the same and the attributes that define them.

Pattern recognition

Abstraction



Filtering out info not needed and generalising info that is needed.



Creating solutions using a series of ordered steps.

Algorithms

Your Friendly CS1010 Lecturers

Mr Tan Tuck Choy, Aaron
CS1010 Coordinator
Office: COM1-03-12
Email: tantc@comp.nus.edu.sg

A/P Tan Soon Huat, Gary
Office: COM2-03-50
Email: gtan@comp.nus.edu.sg

CS1010: Discussion Groups

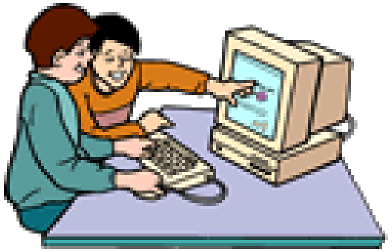
Discussion groups (DGs)

- Small groups (about 15 students per group)
- CS1010 way of calling tutorial groups
- Conducted in labs; 2 hours per week

- Your SG will be pre-allocated.
- Please bid your DG (tutorial group) via CORS during tutorial registration period, i.e.Tutorial Iteration 2 (Round 1A) that starts on 18 Aug 2017 9.00 am.

Workload (4 MCs)

- Lectures:
  - 2 hours/week
- Discussion sessions (tutorials):
  - 2 hours/week in a lab setting.
- Continual assessment:
  - Weekly take-home lab assignments
  - 2 Practical Exams
  - 1 Mid-term Test
  - Final Exam



A Peek at a Lecture Session (1/2)

Instructor’s screen is broadcast to every student’s monitor.

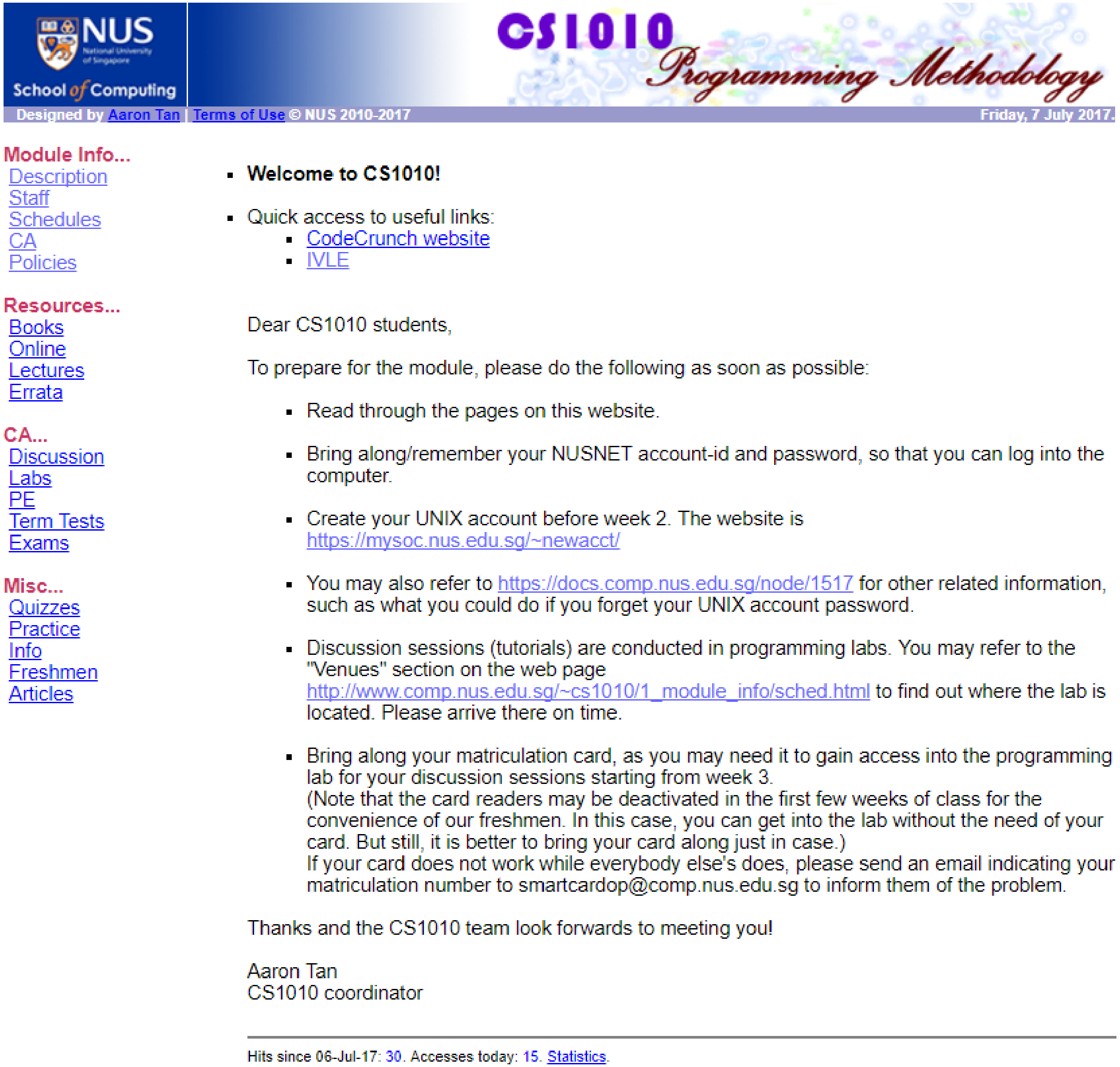
Interacting with students always makes me happy.

A Peek at a Lecture Session (2/2)

Explaining how to edit and compile a program.

Discussing MasterMind.

Module Website http://www.comp.nus.edu.sg/~cs1010


NUS National University of Singapore School of Computing

CS1010 Programming Methodology

Designed by Aaron Tan | Terms of Use © NUS 2010-2017 Friday, 7 July 2017.

Module Info...
Description
Staff
Schedules
CA
Policies

Resources...
Books
Online
Lectures
Errata

CA...
Discussion
Labs
PE
Term Tests
Exams

Misc...
Quizzes
Practice
Info
Freshmen
Articles

- **Welcome to CS1010!**
- Quick access to useful links:
  - CodeCrunch website
  - IVLE

Dear CS1010 students,

To prepare for the module, please do the following as soon as possible:

- Read through the pages on this website.
- Bring along/remember your NUSNET account-id and password, so that you can log into the computer.
- Create your UNIX account before week 2. The website is https://mysoc.nus.edu.sg/~newacct/
- You may also refer to https://docs.comp.nus.edu.sg/node/1517 for other related information, such as what you could do if you forget your UNIX account password.
- Discussion sessions (tutorials) are conducted in programming labs. You may refer to the "Venues" section on the web page http://www.comp.nus.edu.sg/~cs1010/1_module_info/sched.html to find out where the lab is located. Please arrive there on time.
- Bring along your matriculation card, as you may need it to gain access into the programming lab for your discussion sessions starting from week 3.
  (Note that the card readers may be deactivated in the first few weeks of class for the convenience of our freshmen. In this case, you can get into the lab without the need of your card. But still, it is better to bring your card along just in case.)
  If your card does not work while everybody else's does, please send an email indicating your matriculation number to smartcardop@comp.nus.edu.sg to inform them of the problem.

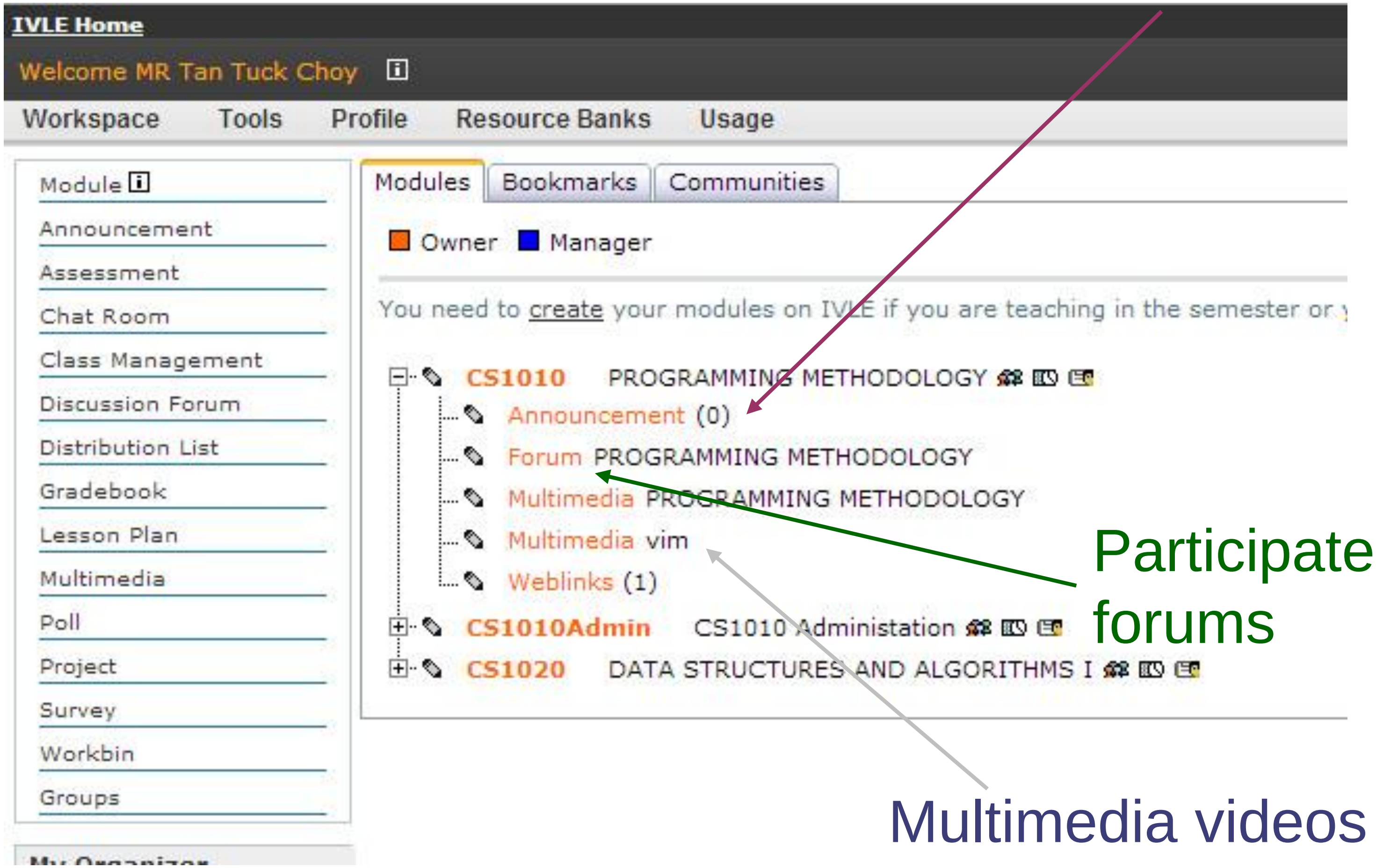
Thanks and the CS1010 team look forwards to meeting you!

Aaron Tan
CS1010 coordinator

Hits since 06-Jul-17: 30. Accesses today: 15. Statistics.

IVLE https://ivle.nus.edu.sg

Watch out for announcements



IVLE Home

Welcome MR Tan Tuck Choy

Workspace Tools Profile Resource Banks Usage

- Module
- Announcement
- Assessment
- Chat Room
- Class Management
- Discussion Forum
- Distribution List
- Gradebook
- Lesson Plan
- Multimedia
- Poll
- Project
- Survey
- Workbin
- Groups

My Organizer

Modules | Bookmarks | Communities

🟧 Owner 🟦 Manager

You need to create your modules on IVLE if you are teaching in the semester or y

- CS1010 PROGRAMMING METHODOLOGY
  - Announcement (0)
  - Forum PROGRAMMING METHODOLOGY
  - Multimedia PROGRAMMING METHODOLOGY
  - Multimedia vim
  - Weblinks (1)
- CS1010Admin CS1010 Administation
- CS1020 DATA STRUCTURES AND ALGORITHMS I

Participate in the forums

Multimedia videos

Mid-Semester Review

Topics in C covered so far

- Basic C program structure
  - main() function
  - Variable declarations
  - Data types (int, float, double, char)
  - Arithmetic operations (+,-,*,/,%)
  - Input/output functions (scanf(), printf())
- Preprocessor directives
  - #include
  - #define
- Control structures
  - Sequential statements
  - Selection statements
    - Relational operators (<, <=, >, >=, ==, !=)
    - Logical operators (&&, ||, !)
    - Conditional operator (? :)
    - Integer as boolean
    - if, if-else, switch
  - Repetition statements
    - while, do-while, for
- Functions
  - Return type
  - Parameters
  - Function prototypes
  - Scope of variables/parameters
- Pointers
- Arrays

Mid-Semester Review

Topics in C

Program development

- ❖ Writing pseudocodes
- ❖ Edit – compile – execute” cycle
- ❖ Step-wise refinement
- ❖ Hand-tracing codes
- ❖ Incremental coding
- ❖ Testing
- ❖ Debugging

Programming environment/tools

- ❖ Operating system: UNIX
- ❖ Editor: vim
- ❖ Debugger: gdb

Problem solving

- ❖ Class exercises
- ❖ Practice exercises
- ❖ Lab assignments

Algorithmic Problem Solving #1: Coin Change

- Given these coin denominations: 1¢, 5¢, 10¢, 20¢, 50¢, and $1, find the smallest number of coins needed for a given amount. You do not need to list out what coins are used.
  - □ Example 1: For 375 cents, 6 coins are needed.
  - □ Example 2: For 543 cents, 10 coins are needed.

Algorithmic Problem Solving #1: Coin Change

Algorithm:

```
    input: amt (in cents)
    output: coins
    coins ← 0
    coins += amt/100; amt = remainder of amt/100;
    coins += amt/50;  amt = remainder of amt/50;
    coins += amt/20;  amt = remainder of amt/20;
    coins += amt/10;  amt = remainder of amt/10;
    coins += amt/5;   amt = remainder of amt/5;
    coins += amt/1;   amt = remainder of amt/1;
    print coins
```

Algorithmic Problem Solving #2: Maximum Sum of Path in a Pyramid

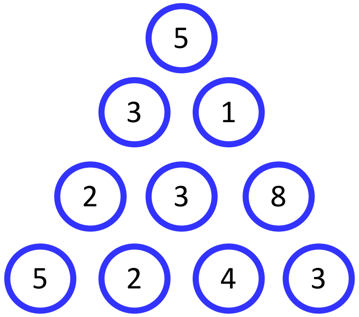
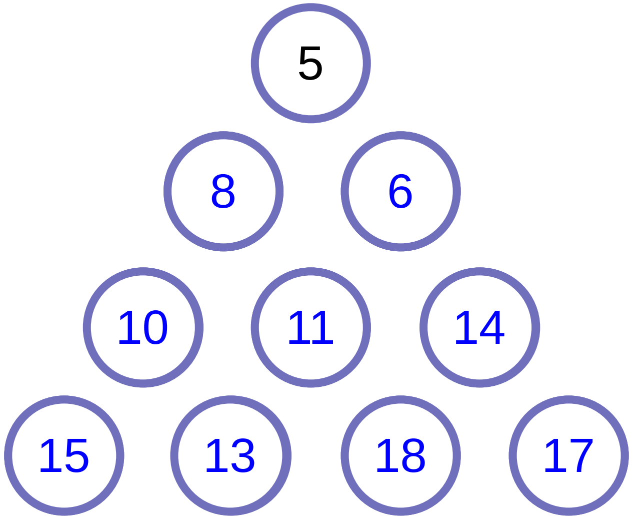
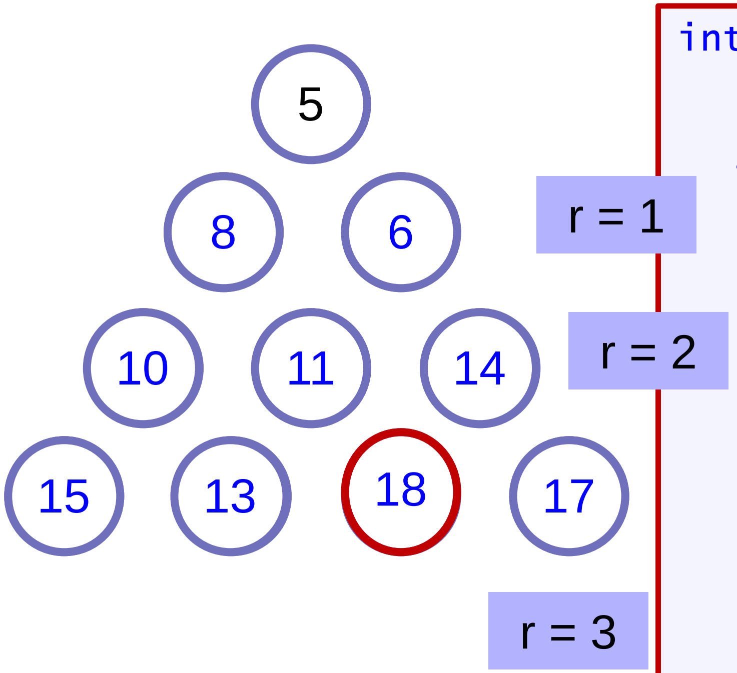


Figure 1. (a) A pyramid of integers. **(b)** A path with sum of 13. **(c)** A path with sum of 18.

Maximum Sum of Path in a Pyramid



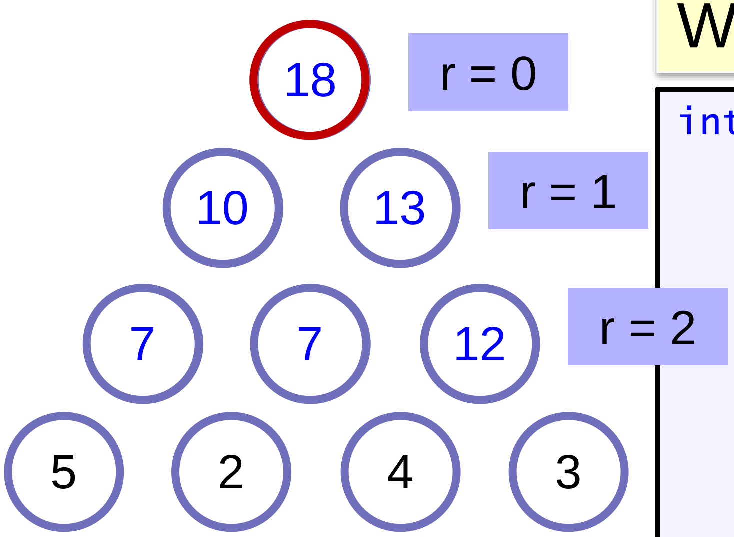
Maximum Sum of Path in a Pyramid



```
int maxPathValue(int arr[][MAX_ROWS], int size) {
   int r, c, max;

   for (r = 1; r < size; r++) {
      arr[r][0] += arr[r-1][0]; // left-most item
      for (c = 1; c < r; c++) {
         if (arr[r-1][c-1] > arr[r-1][c])
            arr[r][c] += arr[r-1][c-1];
         else
            arr[r][c] += arr[r-1][c];
      }
      arr[r][r] += arr[r-1][r-1]; // right-most item
   }

   // find maximum in last row
   max = arr[size-1][0];
   for (c = 1; c < size; c++)
      if (arr[size-1][c] > max)
         max = arr[size-1][c];

   return max;
}
```

Search for largest value in last row.

Maximum Sum of Path in a Pyramid

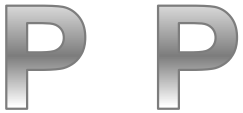


Why not from bottom to top?

```
int maxPathValue(int arr[][MAX_ROWS], int size) {
   int r, c;

   for (r = size-2; r >= 0; r--) {

      for (c = 0; c <= r; c++) {

        arr[r][c] += (arr[r+1][c] > arr[r+1][c+1]) ?
           arr[r+1][c] : arr[r+1][c+1];

      }

   }
   return arr[0][0];
}
```

Algorithmic Problem Solving #3: Mad Scientist

- A mad scientist wishes to make a chain out of plutonium and lead pieces. There is a problem, however. If he places two pieces of plutonium next to each other…

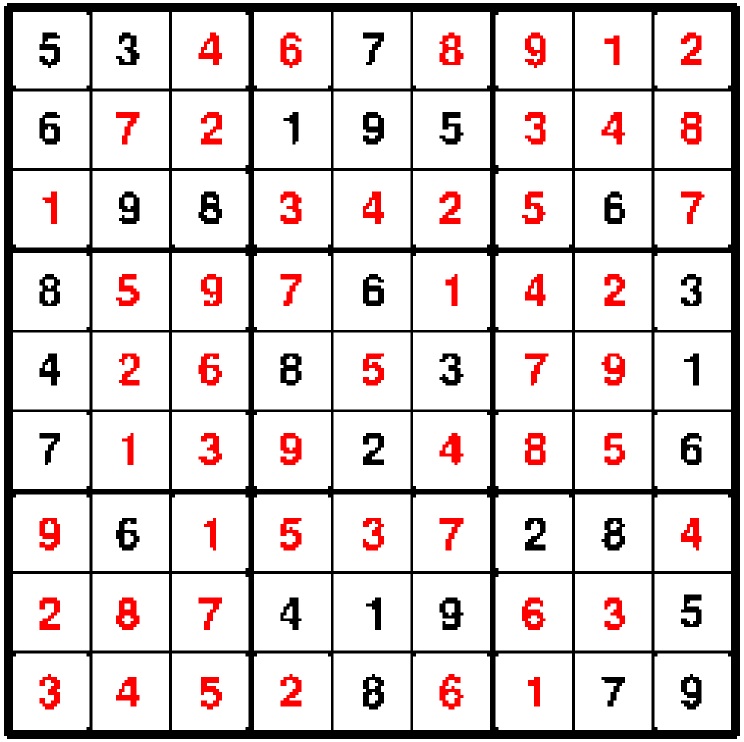


- In how many ways can he safely construct a chain of length **6**?
- General case: What about length *n*?

Algorithmic Problem Solving #3: Mad Scientist

Length	#ways
1	
2	
3	
4	
5	
6	?
n	

Algorithmic Problem Solving #4: Sudoku



5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Algorithmic Problem Solving #5: MasterMind (1/2)

- Sink: Correct colour, correct position
- Hit: Correct colour, wrong position

Secret code	🔵	🟡	🟢	🔴	Sinks	Hits
Guess #1	🟣	🔵	🟢	🩵	1	1
Guess #2	🩵	🟡	🔴	🔵	1	2
Guess #3	🔴	🟡	🟢	🔵	2	2
Guess #4	🔵	🟡	🟢	🔴	4	0

Secret code	🔵	🟡	🔵	🔴	Sinks	Hits
Guess #1	🟣	🟢	🟢	🔴	1	0
Guess #2	🩵	🩵	🟡	🟡	0	1
Guess #3	🟡	🟡	🟢	🟣	1	0
Guess #4	🟢	🟡	🩵	🔵	1	1

Algorithmic Problem Solving #5: MasterMind (2/2)

- 6 colours:
  - R: Red 🔴
  - B: Blue 🔵
  - G: Green 🟢
  - Y: Yellow 🟡
  - C: Cyan ●
  - M: Magenta 🟣
- Given a secret code (secret) and a player’s guess (guess), how do we compute the number of sinks and hits?

CS1010 versus CS1101S

The differences

	CS1010	CS1101S
MC	4	5
Language	Imperative PL (C)	Functional PL ('Homemade' language)
Enrolment	≈ 340 (incl. servicing non-SoC depts.)	≈ 120

The similarities

Small-group teaching
Assume no programming background
Expect students to put in much effort; independent learning

What to Prepare Before Class Starts?

- Check out CS1010 website
  http://www.comp.nus.edu.sg/~cs1010
- Read document “Intro Workshop: Getting Started with UNIX and CodeCrunch) (http://www.comp.nus.edu.sg/CS1010/3_a/labs.html)
  - Learn UNIX
  - Learn vim

Attitude is Everything

- Your attitude, not your aptitude, will determine your altitude.
- If you think you can, you can. If you think you cannot, you are right.
- Don’t complain about heavy workload.
- Work hard, REALLY hard!

We are doing everything we can to help you

- Exercises during discussion sessions
- Practice exercises on CodeCrunch
- On-line quizzes
- IVLE forums
- Help sessions

Announcements

- Choosing CS1101S over CS1010
  - URL: https://register.comp.nus.edu.sg/UGOffice4
  - Deadline: **27 July 2017, Thursday, 6pm**
  - Default: CS1010 (No action needed if you decide to take CS1010)
  - No bidding queue in CORS for both CS1010 and CS1101S
  - Students from IS Dept are precluded from taking CS1101S

DDP students in CS and Maths/Applied Maths will be pre-allocated CS1101S.

For more details, please contact SoC Undergraduate Office @ COM1, Level 2, Room 19

Announcements

This document is available on the CS1010 website

http://www.comp.nus.edu.sg/~cs1010/4_misc/freshmen.html

Have a

GREAT TIME

in

School of Computing!