CS1010 Programming Methodology

A beginning in problem solving in Computer Science

Aaron Tan

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

24 July 2017
Announcements

This document is available on the CS1010 website

Announcements

Choosing CS1101S over CS1010

- URL: [https://register.comp.nus.edu.sg/UGOffice4](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](#)
Ironman draft identifies 18 knowledge areas

- 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

$P = NP$ ?

$O(n^2)$

12 99 37
CS1010

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

**Outcomes**
- Solve simple algorithmic problems
- Write good small programs

**C as a tool**
- Not just about C
Programming

- Language constructs
- Problem solving
- Coding
// 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;
}
Problem Solving Skills

Computational Thinking

Algorithmic Thinking

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

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.

6 July 2017
Computational Thinking

Decomposition

Breaking the problem into smaller, more manageable parts.

Abstraction

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

Pattern recognition

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

Creating solutions using a series of ordered steps.
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

- 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

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.comp.nus.edu.sg/newacc/
- 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 smartcardtop@comp.nus.edu.sg to inform them of the problem.

Thanks and the CS1010 team look forwards to meeting you!

Aaron Tan
CS1010 coordinator
Watch out for announcements

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 \leftarrow 0

coins += amt/100; \text{amt} = \text{remainder of} \ amt/100;
coins += amt/50; \text{amt} = \text{remainder of} \ amt/50;
coins += amt/20; \text{amt} = \text{remainder of} \ amt/20;
coins += amt/10; \text{amt} = \text{remainder of} \ amt/10;
coins += amt/5; \text{amt} = \text{remainder of} \ amt/5;
coins += amt/1; \text{amt} = \text{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

The diagram shows a pyramid with numbers at each level. The goal is to find the maximum sum of any path from the top to the bottom. The numbers in the pyramid are:

- Top level: 5
- Second level: 8, 6
- Third level: 10, 11, 14
- Bottom level: 15, 13, 18, 17

To find the maximum sum, one possible path is 5 → 10 → 15, which sums to 30.
Maximum Sum of Path in a Pyramid

Search for largest value in last row.

```c
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;
}
```
Maximum Sum of Path in a Pyramid

Why not from bottom to top?

```c
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...

  ![ KA-BOOM ]

- 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

<table>
<thead>
<tr>
<th>Length</th>
<th>#ways</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>?</td>
</tr>
<tr>
<td>$n$</td>
<td></td>
</tr>
</tbody>
</table>
Algorithmic Problem Solving #4: Sudoku
Algorithmic Problem Solving #5: MasterMind (1/2)

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

<table>
<thead>
<tr>
<th>Secret code</th>
<th>Sinks</th>
<th>Hits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guess #1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Guess #2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Guess #3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Guess #4</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Secret code</th>
<th>Sinks</th>
<th>Hits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guess #1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Guess #2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Guess #3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Guess #4</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th></th>
<th>CS1010</th>
<th>CS1101S</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MC</strong></td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td><strong>Language</strong></td>
<td>Imperative PL (C)</td>
<td>Functional PL (‘Homemade’ language)</td>
</tr>
<tr>
<td><strong>Enrolment</strong></td>
<td>(\approx 340) (incl. servicing non-SoC depts.)</td>
<td>(\approx 120)</td>
</tr>
</tbody>
</table>

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](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

Have a GREAT TIME in School of Computing!