# Tutorial2:
# Online Judge System

Jiani, ZHANG

CSCI2100a Data Structures

https://www.cse.cuhk.edu.hk/irwin.king/teaching/csci2100/2016

# Assignments

1. Written Assignments
   - Submit a hard copy to the bin box on 10/F SHB before or on due date. Please write your student ID and name on it.
2. Programming Assignments
   - Submit through online judge. Please note that the online judge will be open for only one week for each set of assignments

❖ All the problems will be announced in the course website. Please check the website regularly.

- https://www.cse.cuhk.edu.hk/irwin.king/teaching/csci2100/2016

# 1. Written Assignments

- Usually, the due time will be 5:00 pm.

- Late Penalty: **10 percent** for each day after the due date.

- You can download an excel file from tutorial page four days after due date to check whether we have received your assignment. If there is any mistake, please contact TA as soon as possible.

- Submissions after four days from the due date are not acceptable and will be marked as 0.

# 2. Programming Assignments

- This course adopts **e-mail submission** for programming assignments. We will use our prepared **Online Judge System** to judge your submissions.

- Each assignment will be given **one week** for you to submit (usually from Monday to Sunday). Please do not submit before or after the submission period.

- One problem may be judged by two sets of test cases, a small (simple) one and a large (hard) one, based on the nature of the problem.

# 2. Programming Assignments

- Method to calculate your score:
  - /* Will be adjusted for each assignment */
  - /* For reference only, and does not reflect your final grade */
  - my @award = (**5 , 4 , 3 , 2 , 1 , 0 , 0**); //the k-th entry is the number of award times in the k-th day
  - pen = # of submissions -1-award[x]; //x is the day when you successfully finish the program
  - if (pen<0) pen = 0;
  - if (pen > 12) pen = 12;
  - mark = 100 - pen*5;

pen = # of submissions -1-award[x];
//x is the day when you successfully finish the program

# Example

- e.g: Announcement --*The reward vectors for Problem 1, 2 and 3 are (5,5,4,4,3,2,1), (4,4,3,3,2,2,1) and (5,4,3,2,1,0,0) respectively*.
- Problem award: **(5,4,3,2,1,0,0)**
- 3 students
  - A: 2 failed on 1st day, 2 failed on 2nd day, 1 accepted on 3rd day; **(2+2+1)-1-3=1**
  - B: 4 failed on 1st day, 1 accepted on 1st day; **(4+1)-1-5=-1**
  - C: 1 failed on 1st day, 1 accepted on 7th day. **(1+1)-1-0=1**
- Scores?

# 2. Programming Assignments

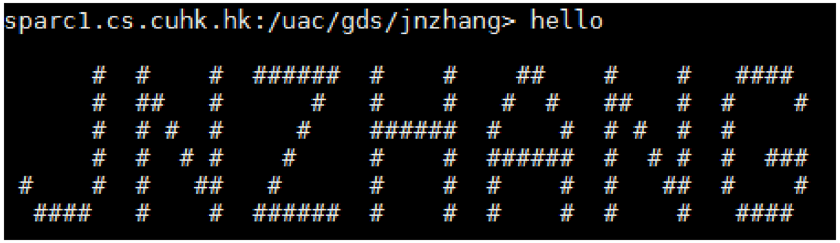
- The ranking for each assignment will be shown during the submission period. The link will be announced later in the course website. The students will be ranked first according to number of problems solved, then total score, finally student ID.
- The ranking page will be updated once someone submits.
- The ranking is for your reference only.

- The most important is to write your **OWN** code.

# Account Preparation

1. SSH to any CSE UNIX environment (e.g. sparc1), at your HOME directory, type the command: *[/uac/cact/csci2100a/bin/setup_csci2100a](/uac/cact/csci2100a/bin/setup_csci2100a)*
2. Then **logout** and **login again**.
3. Test your environment by running the command: *hello*
   A big banner of your username should be shown.

   My example:



```
sparc1.cs.cuhk.hk:/uac/gds/jnzhang> hello

      #  #    #  ######  #    #    ##    #    #   ####
      #  ##   #      #   #    #   #  #   ##   #  #    #
      #  # #  #     #    ######  #    #  # #  #  #
      #  #  # #    #     #    #  ######  #  # #  #  ###
 #    #  #   ##   #      #    #  #    #  #   ##  #    #
  ####   #    #  ######  #    #  #    #  #    #   ####
```

4. Windows(http://mobaxterm.mobatek.net/) / Mac (Command Line)

# Miscellaneous

- If you are not in the CSE network, you can use SSH to connect your CSE UNIX account by following steps.

1. Open SSH/mobaxterm, click the Quick Connect:

   Host Name: *gw.cse.cuhk.edu.hk*

   User Name: *your UNIX account name*

   Mac user (command line):

```
ssh -oHostKeyAlgorithms=+ssh-dss username@gw.cse.cuhk.edu.hk
```

2. Type in your password.
3. Then enter command : *ssh sparc1*
4. Type in your password again.

Now you successfully login to the CSE UNIX platform.

The document will be put on the course website for your reference.

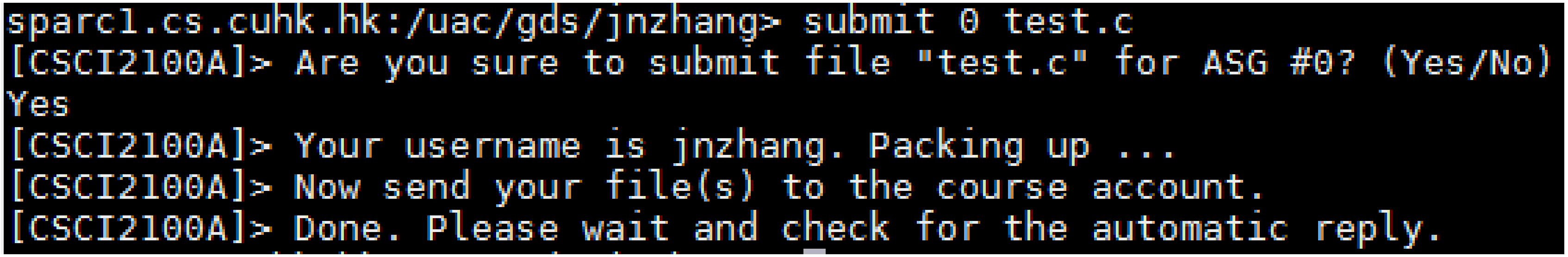
# How to Submit

1. Upload it to your CSE UNIX account.

2. SSH to any CSE UNIX environment, submit the source code by

*submit <asgn no.> <file_name>.c*

where *asgn no.* is the assignment ID and *file_name* is the name of the source code you wish to submit.



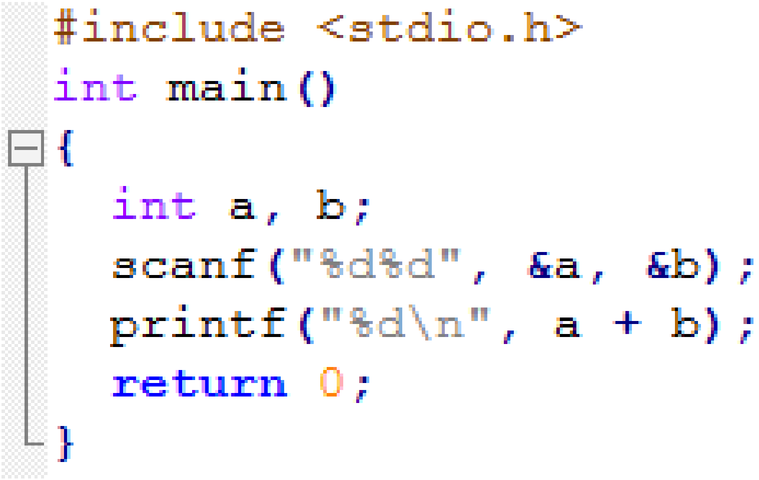
```
sparc1.cs.cuhk.hk:/uac/gds/jnzhang> submit 0 test.c
[CSCI2100A]> Are you sure to submit file "test.c" for ASG #0? (Yes/No)
Yes
[CSCI2100A]> Your username is jnzhang. Packing up ...
[CSCI2100A]> Now send your file(s) to the course account.
[CSCI2100A]> Done. Please wait and check for the automatic reply.
```

3. When you are prompted, double-check the assignment No. and the filename, then type Yes to commit the submission.

4. Upon successful submission, you will receive a judge reply very soon (at your CSE mail account). You should KEEP your reply for future references.

# How to Test

- Assignment 0 is a test for you to try out the submission system. The input contains two integers and you should print out the sum of the integers at your output.
- A sample program is given:



```
#include <stdio.h>
int main()
{
    int a, b;
    scanf("%d%d", &a, &b);
    printf("%d\n", a + b);
    return 0;
}
```

You can check the ranking here.

# Miscellaneous

NOTE: Unless otherwise specified, we have the following configurations for ***all*** tasks:

- Time Limit: **1s**
- Memory Limit: **32MB**
- Output Limit: **32MB**
- Compiler Flags: **-lm -w -O2 -static**

ATTENTION: For those who still do not receive CSE account, we will distribute a temporal CSE account for each of you when add/drop period is over.

# Reply Messages

- **Accepted**
  - Congratulation
- **Wrong Answer**
  - Check your algorithm
  - You may create your own test cases
  - Probably extra spaces or blank lines in the output
- **Compile Error**
  - Please first compile successfully in your own computer
- **Time limit Exceeded**
  - Check your algorithm. Infinite loops?
- **Output Limited Exceeded**
  - Check your algorithm. Infinite loops?

# Reply Messages

- **Restricted Functions**
  - Have any code which may threat the system
  - Any system functions which are not needed should not be called
  - See course webpage for more details
- **Runtime Error**
  - Maybe forgot to “return 0” in main
- **Segmentation Fault**
  - Check your output, make sure it exactly fits the requirements

- ATTENTION: Please make sure that you submit to the right problem

# Reply Message

- Make sure your output is **EXACTLY** the same as the specification, (or sample program, if provided).
  - Check for upper case / lower case letters.
  - Check for extra spaces / extra blank lines.
    - There should be no extra spaces at the end of a line, and no extra blank lines at the end of the outputs.

# Mid-term (Programming) Environment

- The operation system will be Ubuntu Linux 12.04 LTS, with **GUI**
- And the computer configuration will have these basic editors:
  - Vi/VIM
  - Emacs
  - Gedit (with GUI)
  - Nano
- How to debug?
  - Compile your source code using gcc or use gdb

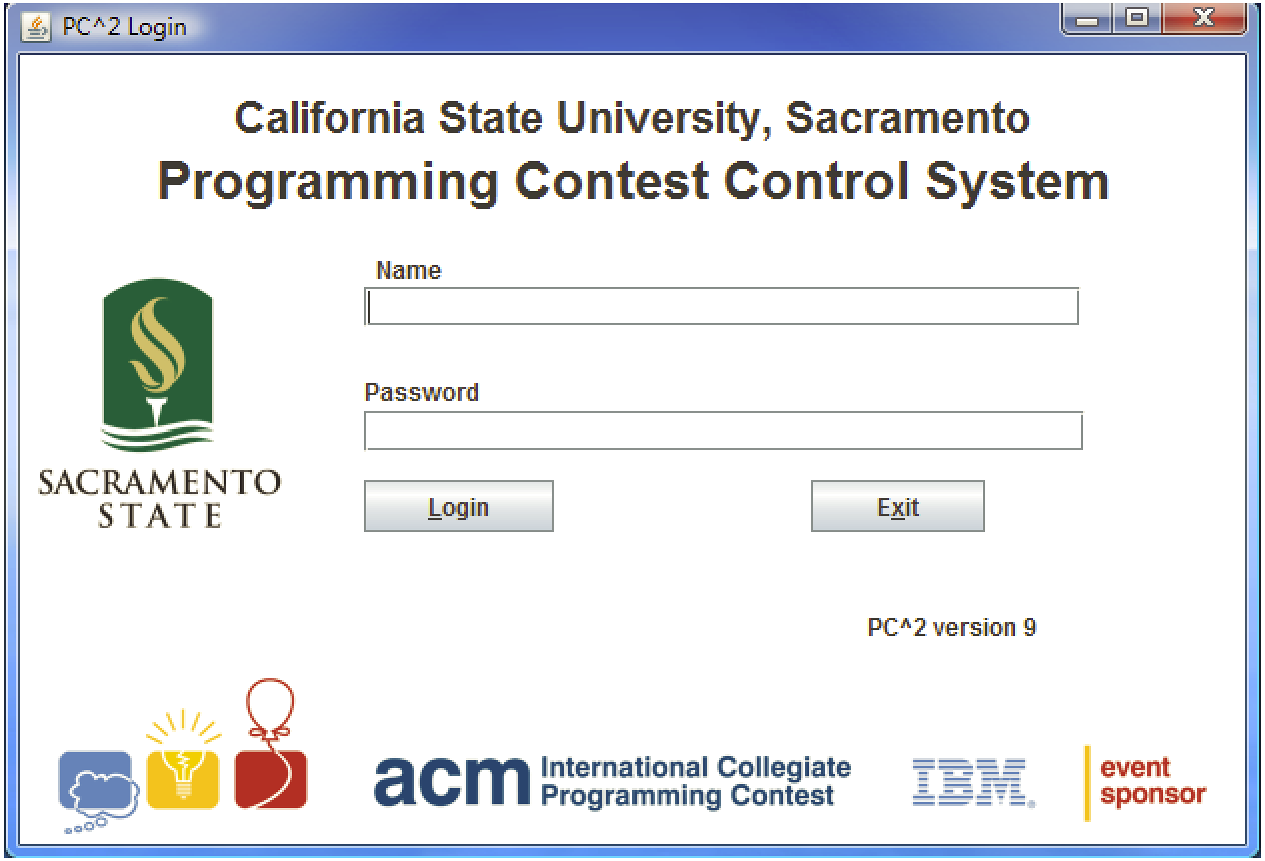
# Submit Tool

- We will use a GUI base tool to help to submit your program —— PC^2 (ACM-ICPC official tools)

- It is simple and robust.

- All things can be done by just several clicks.

# login to PC^2



PC^2 Login

California State University, Sacramento

Programming Contest Control System

Name

Password

PC^2 version 9

# Thanks!

Q & A