CS 101: Computer Programming and Utilization

01-Introduction

Instructor: Sridhar Iyer
IIT Bombay
Course Information

- **Course page:** [http://www.cse.iitb.ac.in/~cs101/](http://www.cse.iitb.ac.in/~cs101/)
  - Long-term information will appear here

- **Moodle page:** CS 101-2013-2
  - [http://moodle.iitb.ac.in/course/view.php?id=3403](http://moodle.iitb.ac.in/course/view.php?id=3403)
  - Day-to-day information will appear here

- Fill out survey when survey link appears
- Find out your lab batch – Will be assigned by Fri
- Lab starts next week – Monday, 13\textsuperscript{th} Jan
Course Resources

- **Textbook:** Introduction to Problem Solving and Programming Through C++ (draft) by Prof. Ranade
  - Link to softcopy will be posted on moodle

- Previous years' course pages.
  - www.cse.iitb.ac.in/~cs101/...

- Other resources: www.cplusplus.com
Places and times

• Lectures (two sections)
  • Slot 11: Tue, Fri 3:30-5:00; 241 students
  • Slot 6: Wed, Fri 11—12:30; 216 students
  • Combined classes may be held on some saturdays; these will be announced in advance

• Labs (OSL: Ground floor, Maths Bldg)
  • Mon-Fri 8:30-10:30 pm (85 students per batch)
  • Makeup lab - Sat- 8:30 -10:30 pm
  • Labs start next week; You will receive mail

• Tutorial
  • This is a 6 credit course. We already have 3 hours of lecture and 2 hours of lab. So no need for tutorials.
  • Tutorial slot will be used if I want a makeup class or if you want a clarification session; will be announced in advance
“The management”

**Instructor**

- **STA**
  - **JTAs**
    - **Mon lab. batch**
    - **Tue lab. batch**
    - **Wed lab. batch**

- **STA**
  - **JTAs**
    - **Thu lab. batch**

- **STA**
  - **JTAs**
    - **Fri lab. batch**

**Web site STAs**

**Assignment STAs**

*cs101stas@cse.iitb.ac.in*

*cs101jtas@cse.iitb.ac.in*

_Do not use personal email_
Assessment

12 % : Quiz 1
25 % : Midsem
12 % : Quiz 2
35 % : Endsem
08 % : Lab assignments
08 % : Lab Project

- There may be minor changes in weightage %, depending on how the labs progress
Lab Assignments

- Announced before the session.
- You may discuss assignment, but code individually.
- Lab assignments are meant more for you to practice than for us to grade you.

Next week:
- how to log in,
- how to use an editor to write a program,
- how to compile the program and run it.
- General information about Unix.
What is this course all about?

- Programming concepts - How to represent problems and solve them using a computer
- Programming language - C++

- No prior knowledge necessary

- Show of hands: How many of you have
  - Taken Java or C++ as an elective in 10th or 12th std?
  - Some other exposure to programming?
  - No exposure?
Why should you care?

• Computers are used in almost every field
  • You are likely to need more than user-level familiarity with application software in your field
  • This course will start you towards understanding how such software are designed and implemented

• Algorithmic thinking (designing solutions in a systematic and step-wise manner) is a useful skill in any discipline
  • Designing and implementing computer programs is a natural way to refine your algorithmic thinking skill
PC building blocks: motherboard

- CPU with cooling fan
- Fast electronic memory
- Magnetic disk data connectors
Storage and peripheral devices

- Display
- Keyboard
- Rotating magnetic platters
- Record/play head on arm
- Data cable between disk and motherboard
Basic capabilities of a Computer

- Ability to handle numerical values
- Ability to carry out numerical operations
- Ability to collect values from as input, and to give back the calculated results, as output
- Ability to store these values temporarily
  - Notion of a “memory” location
  - Ability to refer to locations by symbolic names
A simple program in English

1. Get a number from input, and store it in a location. Call this location A.

2. Get another number from input, and store it in another location. Call this location B.

3. Multiply the numbers in locations A and B, and store the resulting value in yet another location. Call this location P.

4. Give the value stored in location P, as output.
Conceptual understanding using Mr. Dumbo

- See slides 15 to 30 of Prof. Phatak's session 1, from Autumn 2011 offering,

Simplified abstract view

Program that tells the CPU what to do and how to do it

CPU

Address

RAM location 0
RAM location 1
RAM location 2
...

Data

Reserved for display

Reserved for keyboard

Random access memory (RAM)
Rules of the game

- A register or a RAM location can store exactly one value at any time
- Writing into a register or RAM location (or reading into it from the keyboard) destroys the earlier value
- Outputting a register or RAM location to the display is done by copying to the area reserved for the display; this does not destroy the source value
- Accessing any RAM location is equally fast
E.g.: Centigrade and Fahrenheit

- \( C/5 = (F-32)/9 \)
- Painful to say “register 5” and “RAM location 43” so we will use shorter names
- Load input \( F \) from keyboard into R0
- Load 32 into R1
- Store R0–R1 in R2
- Load 9 in R3
- Divide R2 by R3 and store in R4
- Load 5 in R5
- Multiply R5 by R4 and store in R6
- Output R6 to the display
High level programming language

- Input F from keyboard
- Set C to 5*(F-32)/9
- Output C to display

- Programmer need not keep mental map of which register and RAM location contain values of what symbolic variables
- Compiler or interpreter automatically translates from above format to register and RAM location manipulations we saw
Saying it in C++

```c++
main() {
    float fahrenheit;
    cin >> fahrenheit;
    float centigrade = 5*(fahrenheit-32)/9;
    cout << centigrade;
}
```

- Some details omitted
- Save above source code to text file
- Compile source code to executable file
- Call executable from a shell command line
Saying it in Scratch

Start

Get input

Calculate

Give Output

**when clicked**

**ask** Give the Fahrenheit value and wait

**set** f\(\downarrow\) to **answer**

**set** c\(\downarrow\) to \(5 \times f - 32 / 9\)

**say** join The Centigrade value is c
Why Scratch?

• More fun than delving into C++

• For those of you who have no prior exposure
  • We can first focus on the programming concepts, rather than C++ syntax
  • Transition from Scratch to C++ should be easy

• For those of you who have done C++ programming
  • Here is something different that you can learn
  • Have fun creating games