CSB101: Problem Solving And Computer Programming

Module 1: Introduction to Computers

Dr. Chandra Prakash
Assistant Professor
Department of Computer Science and Engineering

Course Objectives

- To develop the basic programming skills in students
- Introductory course that covers how to code using C
- To provide the students with a foundation in computer programming
- To improve their proficiency in applying the basic knowledge of programming to solve problems related to their field of engineering
- Develop attitude to propose new algorithms
- Programming and implementation practice
- Presentation practice
- Good programming practices

Syllabus

- Module 1: Introduction to the Computers
- Module 2: Introduction to programming language: basic of C
- Module 3: Function
- Module 4: Arrays, strings, Pointers
- Module 5: Structure, File Management in C
- Module 6: Graphics in C [Extra]

Reading Material

- Use any standard textbook on ANSI C
- Books
  - Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
  - Yashwant Kanetkar, Let Us C
  - Computer programming and fundamentals in C by Rama Tanjus
  - Kernighan & Ritchie, The C Programming Language

Course website

- Website (slides, important updates)
  - https://cprakash86.wordpress.com/csb101-problem-solving-computer-programming/
- Discussion (lecture related, doubts)
  - cprakash@nitdelhi.ac.in
  - Teaching Assistant [TA]:
    - Shailza Kanwar
    - Jasmine
    - Arvind
**Course Policies**

- Lecture notes, programming assignments, and other useful information will be posted on the course web page.
- You should check the web page regularly.
- The lecture notes will be in powerpoint.
- Discussion of the programming assignments is allowed and encouraged. However, each team is expected to do its own work.
- Assignments which are similar will receive a zero.
- Regular attendance is highly recommended. If you miss a class, you are responsible for all material covered or assigned in class. Late programming assignments will be penalized 10% of the points assigned per day (weekends count as one day).

**Course Components**

- **Continuous Assessments [25]**:
  - Attendance
  - Surprise Quizzes
  - Project
  - Presentation
- **Mid Term [25]**
- **End Term [50]**
- **Lab**:
  - Programming assignments
  - Lab Tests

**About ‘the’ Course**

- An assignment based course
- More emphasis on problem solving
- Programming in C
- Exposure to Linux
- Good programming practices

**Tools for Programming**

- JSLinux is an online terminal to run C programs.
- emacs is a powerful text editor.
- gvim is another powerful text editor.

**Module 1: Introduction to the Computers**

- Part 1.1: Introduction
  - Hardware and Software
- Part 1.2: Basic Model of Computation
- Part 1.3: Algorithm and Flowcharts :
  - Part 1.4: Number System

**Lets Start**
Introduction To Computers

- Computers:
  - An electronic device that is designed to accept data, perform the required mathematical and logical operations at high speed, and output the result.
  - Common in almost all aspects of our daily lives.
- Hard to imagine a world without them.

Computers

- Physical device that implements a computation.
- Determining something by mathematical or logical methods.
- Analog computer: inputs, outputs and internal values may assume real values.
- Digital computer: inputs, outputs and internal values drawn from finite sets.

Computer Vision Problems

- Face Recognition
- Style Transferring
- Image quality enhancement
- Beautification
- Object detection (Self driving car)
- Classification
- Gesture Recognition

Sequence Modeling Problems

- Classification

Programming – Why?

- Computers are used for many different purposes in many different situations.
  - But, how can they be so versatile?
  - Answer: They can be programmed.
- The ability for a computer to be programmed allows it to do whatever their programs tell them to do.
- A program is a set of instructions that tell a computer what to do.
- A computer cannot do anything unless it has a program to tell it what to do.
- In this class, we will focus on writing these programs.

Programming – What?

- Programs are used to operate the components of a computer, solve problems or satisfy a want/need.
  - How long will it take me to get home if I drive x miles per hour?
  - I want to be able to tell my friends what I am doing right now.
- Computer Programming is both an Art and a Science.
  - Every aspect of a program must be carefully designed.
- As an art, programming takes creativity and problem solving.
  - There is no one correct way to solve a problem.
  - As a science, there are formal and proven methods to go about creating a program.
- In this course, you will learn both the art and science of programming.
Programming How??

- Language
- Need Of Language
  - Communication
- Programming Languages

Hardware and Software

- Programs can also be called software.
  - Software refers to the computer programs that a computer uses to complete a task.
- Hardware refers to the physical components that a computer is made of.
  - Each device plays a part.
  - Major components
    - Central Processing Unit
    - Memory
      - Main Memory
      - Secondary Storage Devices
      - Input Devices
      - Output Devices

Hardware in Personal Computer

1. Monitor
2. Motherboard
3. CPU (Microprocessor)
4. Primary storage (RAM)
5. Expansion cards
6. Power supply
7. Optical disc drive
8. Secondary storage (Hard disk)
9. Keyboard
10. Mouse

Central Processing Unit (CPU)

- The CPU is the heart and brain of the computer.
- The CPU continuously does the following things:
  1. Fetch an instruction
  2. Follow the instruction
  3. Produce some resulting data
- The CPU has two parts:
  - Control Unit
    - Coordinates the computer's operations
    - Determines where to get the next instruction
    - Regulates the other major components of the computer
  - Arithmetic and Logic Unit (ALU)
    - Designed to perform mathematical operations
- Speed of CPU mainly depends on:
  - Clock frequency (Hz)
  - Amount of cache memory

Memory

- Data of various types, such as numbers, characters, are encoded as series of bits and needed to be stored in consecutive memory locations.
- A storage area
- To store instructions and data, either temporarily or permanently
  - subsequent retrieval of the instructions and data can be possible on demand.
- Data are stored in memory as binary digits, called bits.

Main Memory

- Main memory holds information that the CPU needs to access quickly.
  - Usually, the instructions to be executed.
- When a program is running, some or all of its instructions are in main memory.
- Memory is divided into sections called bytes that hold equal amount of data.
- Each section is made up of 8 bits.
- A byte is the most basic unit of information a computer can hold. It is a switch that is either on (1) or off (0).
- Each byte is assigned and can be accessed by its address.
- A Memory Address is a unique identifying number associated with a byte in memory.
- Main memory typically is volatile.
  - Volatile Memory – memory that when it loses power, the contents are erased.
Secondary Storage

- **Secondary Storage** is memory that can hold data for a long period of time.
- Programs are usually stored in secondary storage and loaded into main memory as needed.
- Common forms of secondary storage:
  - Hard Drive
  - Disk Drive
  - Removable Storage
  - Optical Disk
  - CD-ROM
  - USB Drive
- Other files can be stored in secondary storage:
  - Documents
  - Pictures
  - Whatever else you save on your computer.

Memory

- Operations common to both primary and secondary memory devices. These are as follows.
  - Read: During this operation, data is retrieved from memory.
  - Write: In this operation, data is stored in the memory.
  - Using read and write operations, many other memory-related functions such as copy and delete are carried out
- **Unit of memory**
  - Kilobyte (KB) = 1024 bytes
  - Megabyte (MB) = 1024 Kilobytes
  - Gigabyte (GB) = 1024 Megabytes
  - Terabyte (TB) = 1024 Gigabytes
  - Petabyte (PB) = 1024 Terabytes
  - Exabyte (EB) = 1024 Petabytes
  - Zettabyte (ZB) = 1024 Exabytes
  - Yottabyte (YB) = 1024 Zettabytes

Input Devices

- **Input** is any data the computer collects from the outside world.
- An **Input Device** is anything that collects data and sends it to the computer.
- Common Input Devices:
  - Keyboard
  - Mouse
  - Scanner
  - Digital Camera
  - Disk Drive
  - USB Drive

Output Devices

- **Output** is any data the computer sends to the outside world.
- An **Output Device** formats data and presents it to the outside world.
- Common Output Devices:
  - Monitor
  - Printer
  - Headphone
  - Projector
  - Disk Drive
  - USB Drive

Devices: Input and Output

Block diagram of Computer
History of Computers

- Extremely large in size
- Required an entire room for installation
- Consume enormous amount of power and too expensive to be used for commercial applications
- Limited tasks such as computing trajectories for astronomical or military applications.
- But now size of computer became smaller and their energy requirement reduced immensely
- Almost all devices today can work with computers
  - Cellular phones, global positioning system (GPS) units, portable organizers, automatic teller machine (ATM), gas pumps etc.

Characteristc Of Computers

- An electronic device that perform a function based on a given set of instructions known as a program.
- A computer accepts data, process it and produces information.

DIKW pyramid
- DATA: raw facts/figures
- Information: processed data
- Knowledge
- Wisdom

eg - 12-12-1992
- Dob : it is fact.
- above 18 year : information

Characteristic Of Computers

- Speed
- Accuracy: (GIGO) garbage in garbage out
- Automation
- Diligence - never get tired of repetitive task
- Versatile - flexible
- Memory
- NO IQ
- Economical: how?
  - mail: important document to send - take 2-3 days
  - email: instantaneously

Generation Of Computers

- Generation in general indicates a state of improvement in the product development process.
- With each new generation of computers:
  - the circuit become smaller and more advanced than that used in the previous generation.
- Focus of every new generation has been on miniaturization, speed, power and efficient computer memory.

First Generation Of Computer: Vacuum Tubes (1940–1956)
- Fifth Generation Of Computer: Artificial Intelligence (2010 – Present)
**Classification Of Computers**

![Classification Diagram]

**Classification**
- Supercomputers
  - Most powerful and fastest type of computer
  - Used for high-precision calculations, weather forecasting, and nuclear energy
  - National Center for Supercomputing Applications
  - IBM Blue Gene/L
  - IBM Blue Gene/P

- Mainframe
  - Supports a vast number of users simultaneously
  - IBM 3090
  - SF-3090

- MiniComputers
  - Used by businesses
  - Digital Equipment Corporation VAX
  - IBM AS/400

- Microcomputers
  - Used by consumers
  - Desktop computer
  - Laptop
  - Palmtop computer / Digital diary / Notebook / PDA

**Applications of Computers**
- Government
- Traffic control
- Legal system
- Retail business
- Sports
- Music
- Movies
- Travel and tourism
- Business and industry
- Hospitals
- Simulations
- Geology
- Astronomy
- Weather forecasting
- Education
- Online banking
- Industry and engineering
- Robots
- Decision support systems
- Expert systems

**Model of Computation**
- Logic Gate Symbols
- Transistors are electronic switches that may be used to realize gates.

\[
\begin{align*}
& x \lor y \lor z = x \lor y \lor z \\
& x \lor y \lor z = x \lor y \lor z \\
& x \land y = x \land y
\end{align*}
\]
Software

- Software refers to the programs that run on a computer.
- Two main categories (for this class):
  - Operating System (OS)
    - A set of programs that manages a computer’s hardware devices and controls their processes.
    - Most modern operating systems are capable of running multiple programs at once.
    - UNIX, Linux, Mac OS X, and Windows are examples
  - Application Software
    - Programs that make the computer useful for the user
    - Solve specific problems or supply a service
    - Word processors, spreadsheets, databases, etc.
    - This is what we will be developing in this class.

Application Software

- Standalone Applications
  - Must be installed
  - Compatible for single-OS
    - Ex: Chrome, MS-office, VLC etc.
- Web Applications
  - No need to install
  - Independent to OS
    - Ex: Facebook, Gmail, Google Classroom, irctc, ERP,
Algorithm

- A program is composed of algorithm and data.
- The four common ways of representing an algorithm are the step-form, flowchart, pseudo-code, and Nassi-Schneiderman.
- Key features of algorithm:
  - Sequence (also known as process)
  - Decision (also known as selection): if proposition then process
    - if proposition
      - then process
    - else process
  - Repetition (also known as iteration or looping)

Example – Algorithm

- Sequence: To add two numbers
  1. Start
  2. Read the first number as A
  3. Read the second number as B
  4. Set A + B = Sum
  5. Print Sum
  6. Stop

- Decision: To test the equality of two numbers
  1. Start
  2. Read the first number as A
  3. Read the second number as B
  4. if A = B
    - then Print "Equal"
  5. else Print "Not Equal"
  6. Stop

- Repetition: To print the first 10 natural numbers
  1. Start
  2. Set I = 0
  3. While I ≤ 10
    - Print I
    - If I = 10
      - then Print "End"
    - else Set I = I + 1
  4. Stop

Difference

- An algorithm is the step-by-step procedure of how to do a particular task.
- A program is the implementation of this algorithm in code. The code can be in any programming language.

Example – Algorithm

- Write an algorithm to check whether a number given by the user is odd or even.
  1. Start
  2. Print "Enter the number"
  3. Read N
  4. Q = N \div 2
  5. R = N \mod 2
  6. If R = 0
    - then Print "N is even"
  7. Else Print "N is odd"
  8. Stop

Home-Work: More examples on algorithm

- Write an algorithm
  - For interchanging swapping two values
  - To find the largest of two numbers
  - To find whether a number is even or odd
  - To print the grade obtained by a student using the following rules:

<table>
<thead>
<tr>
<th>Marks</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Above 75</td>
<td>A</td>
</tr>
<tr>
<td>50-75</td>
<td>B</td>
</tr>
<tr>
<td>35-50</td>
<td>C</td>
</tr>
<tr>
<td>Less than 40</td>
<td>D</td>
</tr>
</tbody>
</table>
- To find sum of first N natural numbers
Next Time

- Part 1.1: Introduction
  - Hardware and Software
- Part 1.2: Basic Model of Computation
- Part 1.3: Algorithm and Flowcharts
- Part 1.4: Number System

References

- Introduction to Computers and Programming, CS0007: Introduction to Computer Programming