

Introduction

Objectives

1. Introduction to (program) design.
2. Development of problem solving skills.
3. Fundamentals of computing.
4. Fundamentals of programming.
5. Introduction to C++ programming.

Why Study Programming?

Virtually all engineers use computers in some way.

Engineers are responsible for output from programs they use.

It's likely that you'll have to write some programs some time.

Programming skills are needed in other courses.

Being able to program is a valuable job skill.

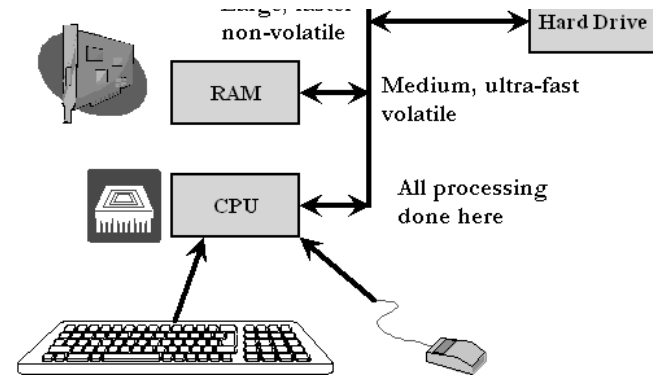
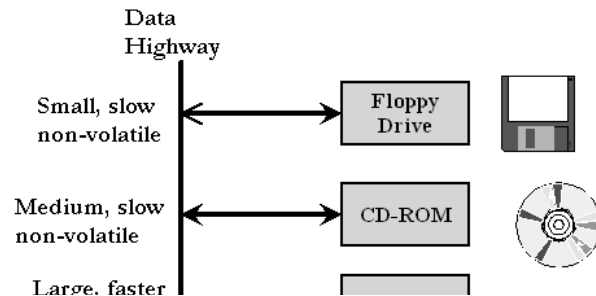
Programming/problem solving skills from this course can be applied to other areas.

It's fun!!

An Introduction to the Computer

What's inside a computer?

Here's roughly the way an engineer would view a computer



The components

1. input devices (mouse, keyboard) which let us input data into a computer
2. output devices (screen, printer) which let us view data.
3. Memory devices, which let us store data, and are subdivided into
 1. volatile memory (RAM) which loses data when the power is off
 2. non-volatile memory (disks of various kinds) which hold data when power is off.
4. The CPU (Central Processing Unit) which processes all data.
5. The Data Highway (Bus) which is used to move data between memory and CPU

When we send data to memory we say we *write* the data

When we fetch data from memory we say we *read* the data.

Computers process data sequentially.

One or more pieces of data are *read* from memory (one at a time)

The data is processed in the CPU

The results are *written* back into memory (one at a time)

Computer Programs

Data processing is carried out using a set of instructions known as a *program*.

The program itself is a form of computer data and is stored in computer memory.

Instructions are *fetched*, one at a time from computer memory to the CPU

Then they are *executed* (carried out) by the CPU.

The Teaching Machine

We are going to use two distinct software sets in this course.

The first (conventional) set is used for creating and building and debugging programs. We will introduce it shortly.

The second is the Teaching Machine which is a tool for demonstrating how computers work.

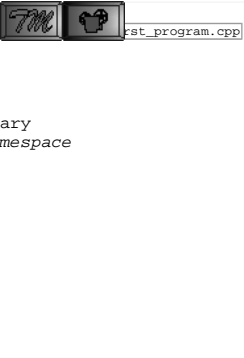
- It is *not intended for creating, testing or debugging* your programs.
- It is intended to show you how a computer processes programs

Here is a first demo of the Teaching Machine

```
/****** First Program *****/
  To demonstrate the teaching machine
  itself
*****/

#include <iostream> // info from standard library
using namespace std; // cout is in the std namespace

int main(){
  int i, j, k;
  i = 4;
  j = -3;
  k = i + j;
  cout << k;
  return 0; // This line is not in the video
}
```



This page last updated on Thursday, January 8, 2004