Computer Science Fundamentals
Intro to Computer Science
– part i 1 –

Behzad Bordbar
School of Computer Science
The University of Birmingham
Module Structure

• Lectures:
  – Tuesdays 09-10, UG04 Learning Centre
  – Tuesdays 16:00-17:00 NG08 Biosciences

• Module Syllabus page
  http://www.cs.bham.ac.uk/internal/modules/2010/21978.html
  http://www.cs.bham.ac.uk/internal/modules/2010/21933.html

Assessment: 100% by exam
Resources

• Module home page: contains the handouts
  – http://www.cs.bham.ac.uk/~bx/b/MainPage/Teaching.html
  – Textbook (for part i1)

• Your Own Notes taken during lectures!

• Lecturer’s office hours
  – Tuesday 10-12 Room 116 Computer Science Building
Content

• Part i1:
  – Introductory overview of computer science using the notion of algorithm as the unifying concept
  – Who knows what is an algorithm?

• Part i2: Computer hardware structure
  - I/O devices and networks
  - Digital logic
  - Operating system
  - Machine code
  - Compilers and interpreters
  - ...
The computer revolution

• Industrial revolution
  – Augmentation of man’s physical powers, amplification of man’s muscle
  – Pressing a button can cause a large machine to stamp a pattern into a metal sheet.
  – Certain repetitive aspects of man’s physical activities were replaced by machines.

• Computer revolution
  – Augmentation of man’s mental powers, amplification of man’s brain
  – Pressing a button can cause a machine to make complex decisions.
  – Certain repetitive aspects of man’s mental activities are being replaced by machines.

• Information technology revolution
  - pervasive access to information via the Internet
Computers and algorithms

- **A computer**
  - is a machine which can carry out **routine** mental tasks by performing simple operations at **high speed**.

- **An algorithm**
  - describes the method how a task is to be accomplished.
  - consists of a finite sequence of steps.
  - which if performed will result in a process being carried out.

- **A processor**
  - carries out a process by executing the algorithm which describes it. *(e.g. a person or a computer)*
# Examples

<table>
<thead>
<tr>
<th>Process</th>
<th>Algorithm</th>
<th>Typical steps in algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>knitting a sweater</td>
<td>knitting pattern</td>
<td>knit one, purl one</td>
</tr>
<tr>
<td>building a model plane</td>
<td>assembly instructions</td>
<td>glue panel A to strut B</td>
</tr>
<tr>
<td>baking a cake</td>
<td>recipe</td>
<td>take three eggs, beat until smooth</td>
</tr>
<tr>
<td>making a dress</td>
<td>dress pattern</td>
<td>sew up side seam</td>
</tr>
<tr>
<td>calculating</td>
<td>formula</td>
<td>$\frac{a}{b} + \frac{c}{b} = \frac{a+c}{b}$</td>
</tr>
</tbody>
</table>
Major hardware components of a computer

A computer is simply a particular kind of processor. Its major components are:

- Central Processing Unit (CPU): performs basic operations
- Memory (MEM): algorithm and data
- Input and output devices (I/O devices): communication
Characteristics of a computer

- Speed
  many millions of instructions per second (rate increases steadily)
- Reliability
  mistakes are due to software not hardware faults
- Memory
  vast, growing steadily
- Cost
  decreases steadily
Example: Airline reservation system

- Many offices, geographically separated by large distances
- Task: combine all seat bookings and cancellations for each flight, disseminate information on seat availability

Performing that task manually would require a massive number of employees and office space. With a computer, however:

- **Speed**: one can continuously maintain accurate status information for each flight
- **Reliability**: passengers can have more confidence in the reservation system than in a manual one
- **Memory**: large quantities of data can be handled
- **Cost**: cost of purchasing and running a computer for the reservation system is far less than employing a large number of clerical workers
Programs and programming languages

• An algorithm must be expressed in such a way that the processor can both understand and execute it (e.g. cook/recipe and musician/score).
• If the processor is a computer then the algorithm must be expressed in a form called program.
• A program is written in a programming language. The activity of writing a program is called programming.
• Each step of an algorithm is expressed by a statement in the program.

A program consists of a finite sequence of statements, each of which specifies certain operations that the computer is to perform.
Who was the first ever programmer?
Who was the first ever programmer?

Augusta Ada Byron, Countess of Lovelace (1815-1852)

Ada Lovelace

Charles Babbage Analytical Engine

…how the engine could be programmed to do various tasks.

Her project was never supported by industry.

Language Ada (successor of pascal…) named after her
Executing a program on a computer

- **Machine languages:**
  - statements can directly interpreted by the computer
  - programming is tedious

- **High level languages:**
  - programs have to be translated to machine language; e.g. C++, Java
The software-hardware hierarchy

- **Applications software** (e.g. packages)
- **System software** (e.g. operating system, language translators)
- **Computer hardware** (e.g. CPU, memory, I/O devices)
Next we look at algorithms

To carry out a process on a computer we must:

1. Design an algorithm which describes how the process is to be performed

2. Express the algorithm as a program in a suitable programming language

3. Get the computer to execute the program

Without an algorithm there can be no program, and without a program there is nothing to execute.

Algorithms are independent of the language in which they are expressed and the computer which executes them.

So, algorithms can be devised and studied independently of the technology of the day!