

Course : CSE 105

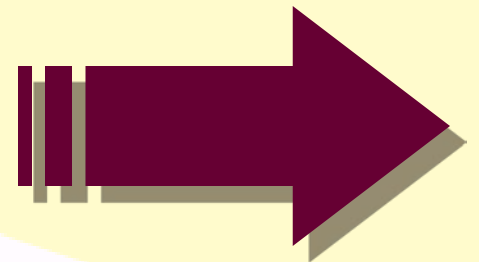
Structured Computer Programming

Khaled Hussain

BSc. Hons (UK), MSc. (UK)

Associate Professor & Dean
School of Science & Engineering
Sylhet International University

**Review
Session**



Assignment 0.99

Find some of the senior students (preferably those who did this course with me) and try to learn about this course!

Recap : What is a Computer Program?

The set of instructions that the computer follows in performing a task.

What is Computer Programming?

Computer Programming is creating programs (software) by writing (coding) a set of instructions that the computer will follow in performing various tasks.

The person who writes programs is called a **programmer**.

Programming Languages

Machine Language

Machine language, is a computer language that is directly understandable by a computer's CPU (central processing unit), and it is the language into which all programs must be converted before they can be run. It consists patterns of bits (i.e., zeros and ones).

Machine code is extremely difficult for humans to read and error-prone.

Machine Language

10100001 00000000 00000000

00000101 00000100 00000000

10100011 00000000 00000000

Assembly Language

Assembly Language is a low level programming language that implements a symbolic representation of the numeric machine codes. It consists of both binary and simple words.

Programming in assembly language are easier than programming in machine language.

MOV AX, A

ADD AX, 4

MOV A, AX

A program called an assembler converts assembly language into the underlying machine language.

High Level Languages

High level languages are basically symbolic languages that use English words and mathematical symbols. These are much user friendly and easier to program with.

```
a = b + c;
```

```
Printf("Hello World");
```

VB.net, Pascal, C, C++, Java are examples of high level languages.

Advantages of High Level Languages

Some advantages of high level languages over assembly and machine language are -

- Machine Independence
- Easy to learn and use
- Fewer errors
- Lower cost
- Built-in library functions
- Easier to maintain

Compiler

A compiler is a computer program that transforms the instructions written in a high-level language (source code) into machine language (object code).

Interpreter

An interpreter reads each line of the source code and converts it to machine code on the fly. This happens every time the program is run. Consequently it is very slow as it converts source code to machine code while the program is running.

Algorithm

An algorithm is an effective method for solving a problem using a finite sequence of instructions. It is a precise step-by-step plan that begins with an input value and yields an output value in a finite number of steps.

Example

1. Start
2. Enter sessional mark
3. Enter midterm mark
4. Enter final mark
5. $\text{total} = \text{sessional} + \text{midterm} + \text{final}$
6. If total is equal or greater than 40 then print "PASS"
7. Else If total is less than 40 but equal or greater than 20 then print "REFERRED"
8. Else print "RECOURSE"
9. End

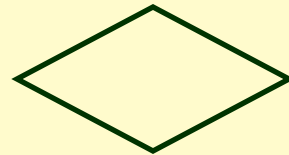
Flow Chart

A flowchart is a pictorial representation of an algorithm that uses different shapes to denote different types of instructions.

Symbols



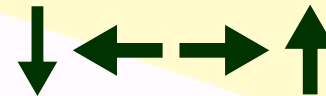
Start / Stop



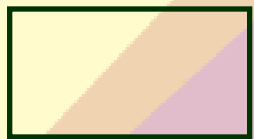
Decision



Input / Output

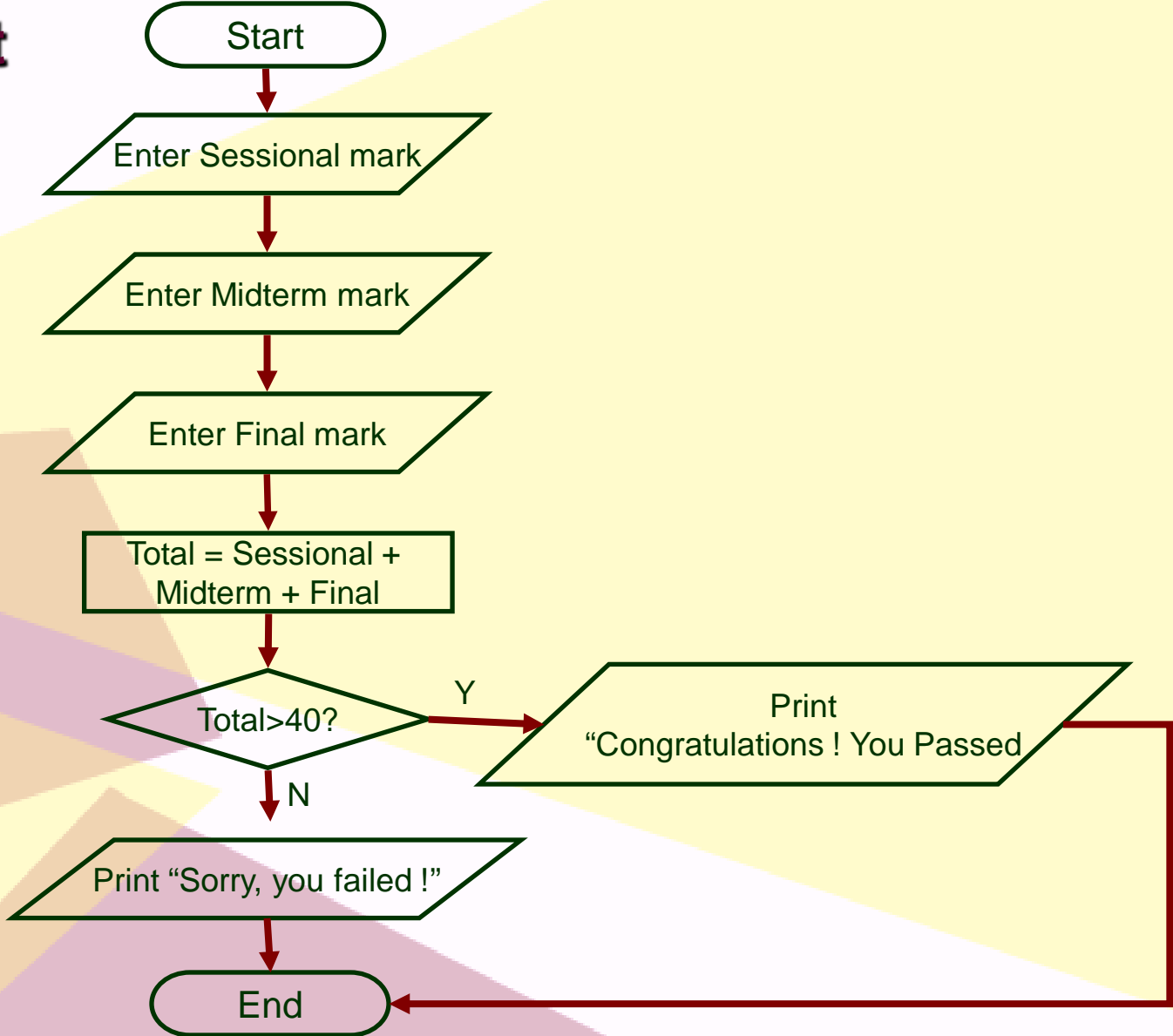


Flow lines



Processing

Flow chart Example



Make sure you understand...

Programming

Machine / Assembly / High Level Language

Assembler

Compiler

Interpreter

Algorithm

Flowchart