An Introduction To Programming

What is Programming?

What is Programming? – 2

- Specifying **what** to to do and **when** to do it

What is Programming? – 3

- Specifying **what** to to do and **when** to do it
- The **what** consists of the following
  - » **At the assembler level the hardwired instructions**
    - > **add, load, store, move, etc.**

What is Programming? – 4

- Specifying **what** to to do and **when** to do it
- The **what** consists of the following
  - » **At the assembler level the hardwired instructions**
    - > **Add, load, store, move, etc.**
  - » **At the Eiffel, C, Java level**
    - > **Assignment, arithmetic, read/write**
    - > **Routines from a Subprogram library, API (Application Program Interface)**

What is Programming? – 5

- The **when** consists of specifying in what order to do the "what" operations

What is Programming? – 6

- The **when** consists of specifying in what order to do the "what" operations
  - » **Control structures**

What is Programming? – 7

- The **when** consists of specifying in what order to do the "what" operations
  - » **Control structures**
    - > **What are the fundamental control structures?**

What is Programming? – 8

- The **when** consists of specifying in what order to do the "what" operations
  - » **Control structures – these are the only ones**
    - > **Sequence**
    - > **Choice**
    - > **Loop**

What is Programming? – 9

- The **when** consists of specifying in what order to do the "what" operations
  - » **Control structures – these are the only ones**
    - > **Sequence**
    - > **Choice**
    - > **Loop**
- What and when are intertwined
  - » **Changing one generally requires changing the other**

What is a program?

- A program is an **algorithm** expressed in a programming language

What is a program?

- A program is an **algorithm** expressed in a programming language
- An **algorithm** is a detailed sequence of actions to perform to accomplish some task.

What is a program?

- A program is an **algorithm** expressed in a programming language
- An **algorithm** is a detailed sequence of actions to perform to accomplish some task.
  - » **Named after an Iranian mathematician Al-Khwarizmi**

What is a program?

- A program is an **algorithm** expressed in a programming language
- An **algorithm** is a detailed sequence of actions to perform to accomplish some task
  - » **Named after an Iranian mathematician Al-Khwarizmi**
- Technically, an algorithm must reach a result after a finite number of steps

An Example Algorithm

- To convert Celsius to degrees Fahrenheit

1. **Multiply the Celsius temperature by 9**
2. **Divide the result by 5**
3. **Add 32 to the result**

Algorithm Key elements

- Notice that there are 3 key elements in this example

Algorithm Key elements

- Notice that there are 3 key elements in this example

1. **What to do – the operation**

Algorithm Key elements

- Notice that there are 3 key elements in this example

1. **What to do – the operation**
2. **What to do it to – the data**

Algorithm Key elements

- Notice that there are 3 key elements in this example

1. **What to do – the operation**
2. **What to do it to – the data**
3. **When to do it – the sequence**

Algorithm Key elements

- The **operations** that can be performed on data are defined for each programming language and associated with special symbols and **keywords**

Algorithm Key elements

- The **operations** that can be performed on data are defined for each programming language and associated with special symbols and **keywords**
- The data elements that can be manipulated fall into two categories
  - » **Literal – data that is a constant value**
  - » **Variable – data that can change value**

Algorithm Key elements

- The **operations** that can be performed on data are defined for each programming language and associated with special symbols and **keywords**
- The data elements that can be manipulated fall into two categories
  - » **Literal – data that is a constant value**
  - » **Variable – data that can change value**
- The order in which the instructions are executed is determined by their **sequence**

Key elements of Programs

- While the example demonstrates several key elements of algorithms it omits an essential element of all programs

Key elements of Programs

- While the example demonstrates several key elements of algorithms it omits an essential element of all programs

Input / Output (I/O)

Key elements of Programs

- While the example demonstrates several key elements of algorithms it omits an essential element of all programs

Input / Output (I/O)

- To be useful a program must accept input data and produce output data

Key elements of Programs

- Input / Output

Key elements of Programs

- Input / Output
  - » **How a program communicates with the user**
- Logic

Key elements of Programs

- Input / Output
  - » **How a program communicates with the user**
- Logic
  - » **The order in which activities are carried out**

Key elements of Programs

- Input / Output
  - » **How a program communicates with the user**
- Logic
  - » **The order in which activities are carried out**
- Data

Key elements of Programs

- Input / Output
  - » **How a program communicates with the user**
- Logic
  - » **The order in which activities are carried out**
- Data
  - » **Its storage and manipulation**

Evolution of I/O

- Early in the history of computing, programs were submitted on punch cards with all the data they required and executed together with other programs that used the same libraries. Output was to a line printer.

- Later developments introduced **interactive processing** which allowed the user to provide data while the program was running. This normally took place in a Question & Answer format.

Evolution of I/O

- The development of the Graphic User Interface (GUI) provided the opportunity for another development

Evolution of I/O

- The development of the Graphic User Interface (GUI) provided the opportunity for another development

- The GUI is not restricted to interacting with the user one line at a time

Evolution of I/O

- The development of the Graphic User Interface (GUI) provided the opportunity for another development

- The GUI is not restricted to interacting with the user one line at a time
  - » **It can present several options to the user and respond to whichever is selected**

Language Paradigms

- The evolution of I/O reflected changes in the way programmers looked at programming

Language Paradigms

- The evolution of I/O reflected changes in the way programmers looked at programming
- Several **paradigms** have been developed

Language Paradigms

- The evolution of I/O reflected changes in the way programmers looked at programming
- Several **paradigms** have been developed
- What is a **paradigm**?

Language Paradigms

- The evolution of I/O reflected changes in the way programmers looked at programming
- Several **paradigms** have been developed
- What is a **paradigm**?
  - » **A set of assumptions, concepts values, and practices that consititute a way of viewing reality.**

Imperative or Procedural Model

- Algorithms are expressed as a hierarchy of tasks.
  - » **Fortran, Cobol, Basic, C, Pascal, Ada**

Functional Model

- Computation is expressed in terms of the evaluation of functions
- A solution is expressed in terms of function calls.
  - » **Functions are both input and output**
  - » **Lisp, Scheme, and ML**

Logic Model

- Based on symbolic logic
  - » **A program consists of**
    - > **A set of facts about objects**
    - > **A set of rules about relationships between the objects**
    - > **A way to ask questions about the objects and their relationships**
  - » **PRO(gramming) in LOG(ic)**
    - > **Prolog**

Object-Oriented Model

- Views the world as interacting objects
- Objects are active and send messages to each other
  - » **SIMULA, Smalltalk, C++, Java, Eiffel**

Event Driven Programming

- Clicking has become a major form of input to a computer

Event Driven Programming

- Clicking has become a major form of input to a computer
- Mouse clicking is not within the sequence of the program

Event Driven Programming

- Clicking has become a major form of input to a computer
- Mouse clicking is not within the sequence of the program
- A user can click a mouse at any time during the execution of a program

Event Driven Programming

- Clicking has become a major form of input to a computer
- Mouse clicking is not within the sequence of the program
- A user can click a mouse at any time during the execution of a program
- The system consists of
  - » **An interface of objects that the user can manipulate**

Event Driven Programming

- Clicking has become a major form of input to a computer
- Mouse clicking is not within the sequence of the program
- A user can click a mouse at any time during the execution of a program
- The system consists of
  - » **An interface of objects that the user can manipulate**
  - » **A collection of routines that will be executed when each event occurs**

Event Driven Programming

- You still have to deal with the thee key elements
  - » **I /O**
  - » **Data**
  - » **Logic**