Lecture 6.1

Topics

1. Comparison Operators – Introduction
2. Statements – Brief

---

1. Comparison Operators – Introduction

There are several comparison operations that are available C. They are given as follows,

```
Greater Than                    >
Greater Than or Equal To        >=
Less Than                       <
Less Than or Equal To           <=
The Same as (or Equal To)       ==
```

For examples

```
iValue > 0
iValue == 0
(iValue % 2) == 0
```

More examples and explanations will be given in class

2. Statements – Brief

In general, there are six types of statements in C (as well as other modern programming languages) and they are

- Expression Statement
- Compound (Block) Statement
- Labeled Statement
- Selection/Conditional Statement
- Iterative (Loop) Statement
- Jump Statement

The last statement, **JUMP SATEMENT**, is available but is discouraged to be used (as well as the **LABEL statement** to go together with the **JUMP statement**).

In forming the logics for the solution (or solution logics), one may have to think about jumping off to another logic as the condition would dictate. However, in implementing or coding, one should avoid any jumps or ill-advised breaks that would lead to broken execution flows; thus to avoid the **Jump statement** as if it is possible (it is always possible in modern programming languages!).

Thus, we will be working with the other five statements during the course of the semester.

We have used **the expression statements** in many examples given in class. We will be working with the **conditional** and **loop** statements (also called structures).

The compound statement (and label statement) will be parts of many of functions, and conditional and loop statements.

2.1 Basic Computer Programming Structures

In general, a programming structure is a basic unit of programming logic.

There are three basic programming structures:

(a) Sequence (or step),

(b) Selection/decision/condition, and

(c) Repetition/loop.

A program may combine these three structures to produce the solution logic for a particular problem.

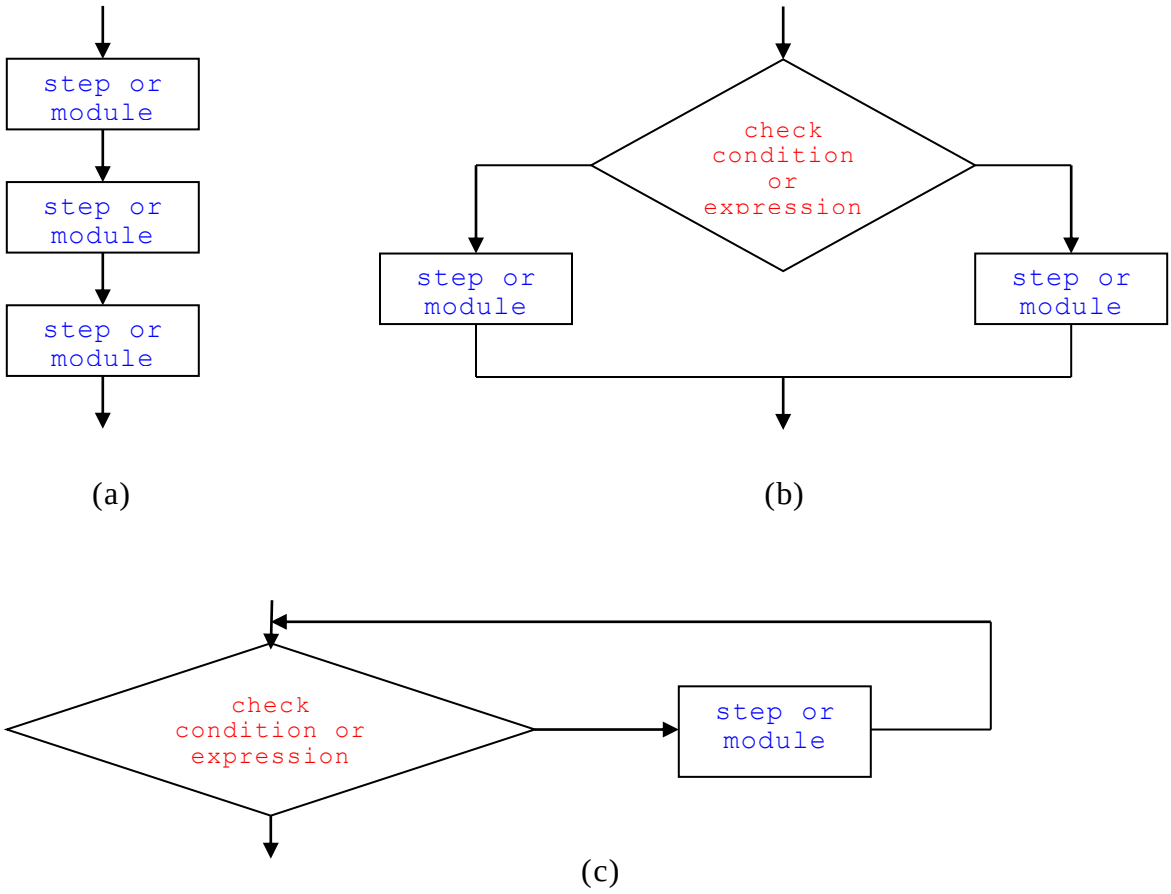

Figure 1 Three basic programming structures

2.2 Sequence (Step) Structure

A sequence structure will provide programming operations or events (for examples, thinking of function calls) in sequence one after another (or step by step).

This is depicted in **Figure 1a**, where each step or module can be just a single operation or several operations (or functions) combined.

2.3 Selection/Decision or Conditional Structure

A selection or decision structure is depicted in **Figure 1b**. There is a set of expressions to be verified before the subsequent operation can be followed.

In the basic structure shown of above in **Figure 1b**, there are only two options to be considered as the outcomes of a decision. If the outcome represents a **true** then the code/execution flow will continue with one path. If the outcome represents a **false** then the flow will follow the other alternative path.

2.4 Loop/Repetition Structure

The loop structure is depicted in **Figure 1c** where the decision is checked before a selected event can be performed or followed.

In C, there are different variations of loop structures (as well as conditional structures). We will revisit them in incoming lectures.