**Practical no:-05**

**Aim :** Modeling Data flow diagram and Control Flow diagram

**Data Flow Diagram :-**

DFD provides the functional overview of a system. The graphical representation easily overcomes any gap between ’user and system analyst’ and ‘analyst and system designer’ in understanding a system. Starting from an overview of the system it explores detailed design of a system through a hierarchy. DFD shows the external entities from which data flows into the process and also the other flows of data within a system. It also includes the transformations of data flow by the process and the data stores to read or write a data.

**Graphical notations for Data Flow Diagram**

| **Term** | **Notation** | **Remarks** |
| --- | --- | --- |
| External entity |  | Name of the external entity is written inside the rectangle |
| Process |  | Name of the process is written inside the circle |
| Data store |  | A left-right open rectangle is denoted as data store; name of the data store is  written inside the shape |
| Data flow |  | Data flow is represented by a directed arc with its data name |

**Levels of Data Flow Diagram (DFD)**

**0-level DFD**

It is also known as a context diagram. It’s designed to be an abstraction view, showing the system as a single process with its relationship to external entities

**1-Level DFD**

This level provides a more detailed view of the system by breaking down the major processes identified in the level 0 DFD into sub-processes. Each sub-process is depicted as a separate process on the level 1 DFD.

**2-level DFD**

This level provides an even more detailed view of the system by breaking down the sub-processes identified in the level 1 DFD into further sub-processes. Each sub-process is depicted as a separate process on the level 2 DFD.

**Control Flow Diagram :-**

A **Control Flow Graph (CFG)** is the graphical representation of control flow or computation during the execution of programs or applications. Control flow graphs are mostly used in static analysis as well as compiler applications, as they can accurately represent the flow inside a program unit.

**Characteristics of Control Flow Graph**

* The control flow graph is process-oriented.
* The control flow graph shows all the paths that can be traversed during a program execution.
* A control flow graph is a directed graph.

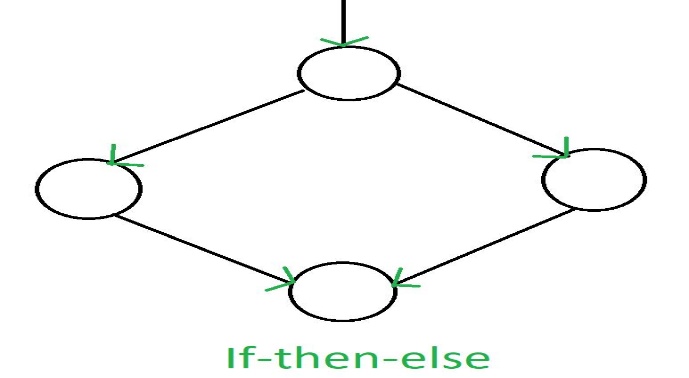
There exist 2 designated blocks in the Control Flow Graph:

* **Entry Block:**The entry block allows the control to enter into the control flow graph.
* **Exit Block:** Control flow leaves through the exit block.

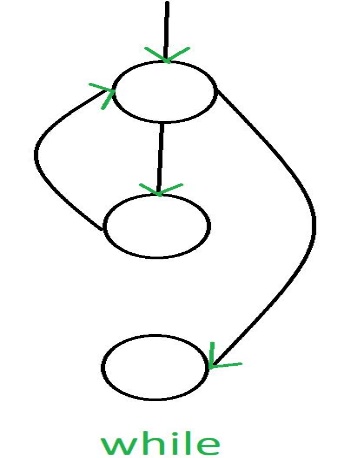
**General Control Flow Graphs**

Control Flow Graph is represented differently for all statements and loops. The following images describe it:

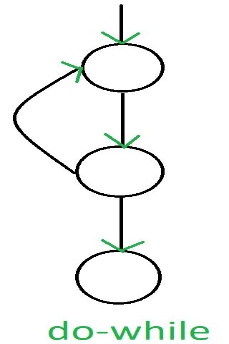
**1. If-else**



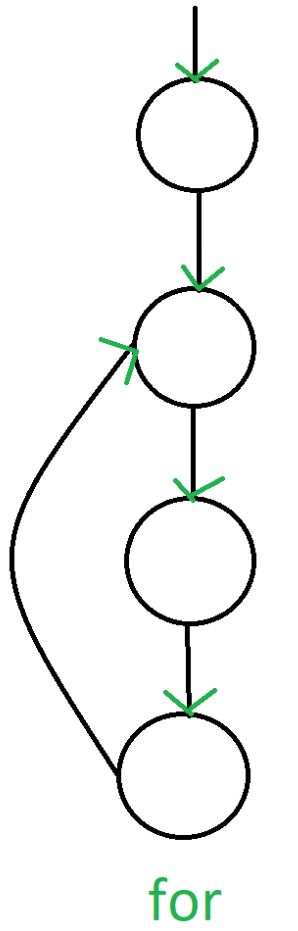
**2. While**



**3. do-while**



**4. for**



* **DFD (Data Flow Diagram)**: Shows how data moves within a system, focusing on inputs, processing, storage, and outputs. It helps in system analysis and design.
* **CFD (Control Flow Diagram)**: Represents the execution flow, decisions, loops, and control logic of a system or program. It is useful for understanding and optimizing program logic.

Both diagrams are essential for system design—DFDs emphasize **data movement**, while CFDs focus on **process control and execution flow**.