

Loops.

looping can be defined as repeating the same process multiple times until a specific condition satisfies

Types of loops.

- 1. do while
- 2. while
- 3. for

do-while loop

- It is used when it is necessary to execute the loop at least once.
- It tests the condition at the end of the loop body.

eg

do E

// code to be executed

I while (condition); jump set after loop starts

↓ do.

3 (→b) Statement(); not (condition) not

break ad at start if

True.

Condition

↓
false.

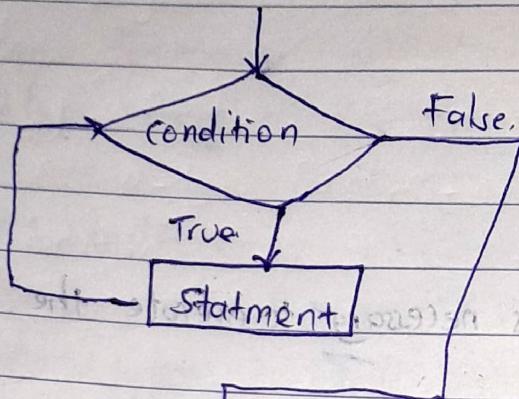
While loop.

- It is used in the scenario where we don't know the no of iterations in advance.
- It tests the conditions before executing the loop

eg

while (Condition) {
 // code to be executed
}

3



final for loop will be -

that goal sets working out to variables with start + -

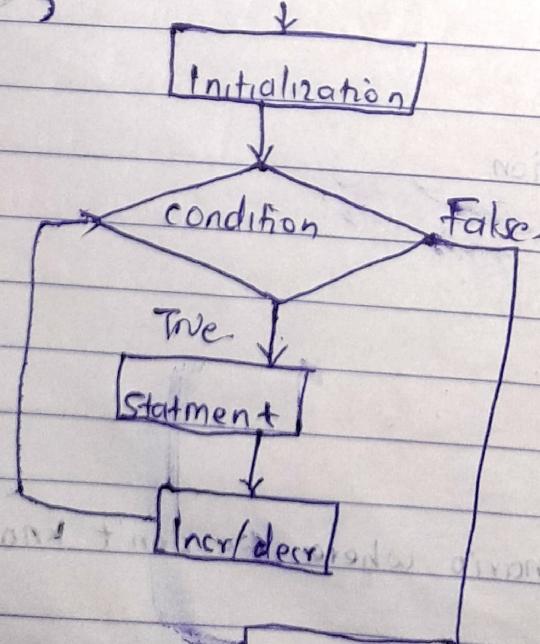
For loop

It is used in the case where we need to execute some part of the code until the given condition is satisfied

eg

for (Initialization; condition; incr/decr) {
 // code to be executed
 }

3



to present work + time complexity with in begin with -

goal sets working out to variables with start + -

Uses of loops

- Loops simplifies the complex problems into the easy ones.
- Enables us to alter the flow of the program so that instead of writing the same code again and again we can repeat the code for finite number of times.

Arrays

Array is defined as the collection of similar type of data items stored at contiguous memory locations.

- Arrays are derived data type in C programming language which can store the primitive type of data such as int, char, double, float.
- Also has the capability to store the collection of derived data types.

Advantages of arrays

Code optimization - less code to access the data.

Ease of traversing -

Ease of sorting -

Random access - we can access any elements randomly using the array

Array declaration and initialization.

data-type array-name [array-size];

e.g.

```
int marks[5];
```

e.g.

```
int marks[5];
```

```
marks[0] = 80;
```

```
marks[1] = 60;
```

```
marks[2] = 70;
```

```
marks[3] = 85;
```

```
marks[4] = 75;
```

Functions.

- We can define a function which can perform multiple tasks.
- A function is a group of statements that together perform a task.
- Every C program has at least one function which is main().

Defining a function

return-type function-name (parameters) body of the function

The return-type is the data type of the value the function returns.

Function name - This is the actual name of the function.

Parameters - It is like a placeholder.

Function body - It contains a collection of statements that define what the function does.

/* Function returning the max between two numbers

int max(int num1, int num2) {

* local variable declaration */

int result;

if (num1 > num2)

result = num1;

else

result = num2;

return result;

}

Difference between do-while loop and while loop.

While

Do while

1. It checks the condition first. 1. The loop will execute and then executes statements. 1. The loop will execute the statement at least once, then the condition is checked.
2. It is entry controlled loop 2. It is an exit controlled loop
3. The condition is mentioned 3. The condition is specified at the starting of the loop after the block is executed
4. It allows initialization of 4. It allows initialization of counter variables before counter variables but after starting the body starting the body of a loop
5. We do not need to add a semicolon at the end of a while condition. 5. We need to add a semicolon at the end of the while condition.

example in a code.

While loop:

```
while (condition) {  
    Statements;  
    // loop body  
}
```

do while loop:

```
do {  
    Statements;  
    // loop body  
} while  
(condition);
```

Structure

- Structure is a user-defined data type that enables us to store the collection of different data types.

- Each element of a structure is called a member.

- struct keyword is used to define the structure.

Example: `struct` is a keyword used to define a structure.

`struct structure_name { data_type member1; data_type member2; ... }`

member1 and member2 are variables of data type member1 and member2 respectively.

data_type member1; is used to declare member1 as data type member1.

data_type member2; is used to declare member2 as data type member2.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

member1 and member2 are initialized to zero by default.

1. Syntax errors

- There are type of errors that occur when code violates the rules of the programming language such as missing semicolons, brackets, or wrong indentations of code.

2. logical errors

They occurs when incorrect logic is implemented in the code and the code produces unexpected output.

3. Runtime errors

This are errors caused by unexpected condition encountered while executing the code that prevents the code to compile.

- Can be null pointer references

4. Time limit exceeded error:

It is caused when a code takes too long to execute and execution time exceeds the given time in any coding contest.

5 Semantic errors:

This are mistakes in the meaning code. eg when you use the wrong variable name or call a function with the wrong arguments

- To avoid you use clear, descriptive variable and function names.

Applications of Arrays.

1. Storing and accessing data - used to store and retrieve data in a specific order
2. Sorting - used to sort data in ascending or descending orders.
3. Searching - can be searched for specific elements using algorithms
4. Single processing - Arrays to represent a set of samples that are collected over time. In applications such as speech recognition
5. Data mining - Arrays are used in data mining application to represent large datasets
6. Robotics - Are used to rep the position and orientation of objects in 3D space.