C Programming:Programming language is a medium to communication with the computer system.

Why we need Programming: To solve complex calciulation fasty we need a programing language.

What is programs: Programs are the set of instructions written in a arranged manner to solve a specific prolem.

Translator: Translator is a medium which is used to convert high level language in machine language or computer understandable code.

Mainly two types of translator are used with computer programming.

1. Compiler: Scan whole program ata a time and displays the list of errors in whole programming.

2. Interpreter : Scans programs line by line . and next line will check only if previous lines are are erroe free.

-----------------------------------------------------------------------------------------------------------------------------------------------

BAsic structure of C Programming:

1. Documentation Section:

Comment: Comment are those lines of code that is not executable by compiler.

1. Single Line: //

2. Multi line: /\*...........\*/

2. Linking Section:

Header files: Header files are the collection of pre-defined functions. whenever you need those functions in your program firstly you have to include the header file in the program.

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<math.h>

#-Pre-Processor directive

Include: folder name

stdio.h :header file.

3. Global declaration Section

4. Main function:

main() // pre-define function

5. Block of main:

---------------------------------------------------------------------------------------------------------------------------------------------------

Input Output function:

1.Keyword: Keyword are some pre-define words that is used to perform any specific task.

Total 32Keywords are define in C programming.

if do, while,int,char,float,double etc.

2.Variables:

variable is atemporary storage area.

a. Declaration of variable:

data\_type variable\_name.

2. Initialization of variables:

a. Direct Input

b.User Input.

Output:

Input function:

scanf():

syntax:

scanf("format\_specifier",variable\_name\_with\_&);

int:%d;

float:%f

double:%lf

long double:%Lf

char:%c

long int:%ld

int a;

scanf("%d",&a)

output function

printf()

printf("format\_specifier",variable\_name)

example:

Operator: Operators are the some special symbols that are used to perform some specific task. Operators are always used with one or more operands.

Operators are the some special symbols that is pre defined in the program and used to perform a pre defined specific task.

accordindg to the working of different operators, that are divided into 7 parts:

1 Arthemetic Operator:

Arthemetic Operator are use to perfom arthmetical operations.

+:addtion

-:Substraction

\*:multiply

/:division

%:modular division

eg.:

void main()

{

int a,b;

}

----------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------

2. Relation Operator:

Relation Operator isUsed to define relation between two oprands.It always returns value either true or false.

<:greater than

>:less than

>=:greater than equal to

<=:less than equal to

==:equals to

!=:not equal to

-------------------------------------------------------------------------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------------------------------------------------------------------

3. Logical Operator:

Logical operator is used to specify ralation between two or more relation.It alwas returns value either true or false.

&& : Logical AND

|| : Logical OR

! : Logical not

Logical AND:

True && True :True

True && False: False

False && True:False

False && False: False

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

True || True: true

True || False: True

False || True:True

False || False:False

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

! True: False

!False: True

----------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------

4. Assignment Operator:

Assignment operator is used to compare is used to right hand side value to the left hand side.

=: is the assignment operator.

int a=40,b;

b=a;

b==a;

----------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------

5. Conditional Operator:

Conditional Opertaor is used as the optional part if-else statement.

Syntax:

condition ?true\_block: false\_block;

eg.:

int a=20;

a>10?printf("a is greater than 10"):printf("a is not greater than 10");

----------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------

6. Increment & Decrement Operator:

Increment & decrement operator is an unary operator. It is used to increase and decrease value of a variable from 1.

++: increment operator

--: decrement operator

----------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------

7. Bitwise Operator:

Bitwise Operator is on binary values.

&: Bitwise AND

|: Bitwise OR

^: Exponential

------------------------------------------------------------------------------------------------------------------------------------------------------------

04/11/2020

Control Statment

1.Conditional Statment:

**if Statment:**

simple if

if-else:

if else statement is used is only one condition is given, but you have to execute different statment when condition is true or condtion

In if-else statement, if given condition is true then block of if will execute, if given condition is false then else block will execute .

…

**syntax**

if()

{

//true statement;

}

Else

{

False statement;

}

Program ta given numer even or odd:

ladder if

else if ladder: else if ladder statement is used when multiple condition is given but only one condition will execute at one time. Then else-if ladder statement is used.

**Syntax:**

If(condition)

{

}

else if (condition)

{

}

else if (condition)

{

}

nested if

**Simple if statement:**

**Simple is statement** is used when you have only one condition, and a number of lines has to execute only if condition is true,

**Syntax**:

If(condition)

{

}

**Wap to calculate percentage of a student**

#include<stdio.h>

#include<conio.h>

void main

|  |  |  |
| --- | --- | --- |
| Wa |  |  |

Wap to find total amount by accepting quantities and rate