

Unit-II

C-programming Basics

Fundamental rules:

About C-programming:

procedural language → Instructions in C pgm are executed step by step

portable → You can move C pgm one platform to another, and run it without ^{any} or minimal changes

Speed → faster than most programming languages like Java, Python etc..

General purpose - It can be used to develop OS, embedded sim, DB and so on.

Features of 'C'.

① → It was a structural language whose rich set of built-in functions & operators can be used to write any complex pgm

② → It is combined both high level and low level languages features

③ → Pgm written in 'C' are efficient and fast.

④ → There are only 32-keywords & its strength lies in its build-in fun.

⑤ High portable:
written for one computer can be run on other with little or no modifications.

⑥ C pgm language is well suited for structured programming, they ~~requiring~~ requiring the user to think of a pgm in terms of fun. modules or blocks debugging (easy), testing, maintenance easier.

⑦ Another important feature of 'C' is its ability to extend itself.

Structure of 'C' Programs:

→ It can be viewed as a group of building blocks called function.

→ A fun. is a subroutine that may include one (or) more statement designed to perform a specific task.

1. document anion section. → one reference
addition // single line comment
include <stdio.h>

2. link section. → instruction to compiler
\$ in library → like

3. definition section # define C [10] r [10]
constant value define. all symbolic constant

4. global declaration section int I, J;
constant value 100
PI 3.14
equal with

5. Main () function section.

{
 declaration part
 executable part
}

6. Sub program selection.

{
 Function 1
 Function 2
}

} user defined fun

Documentation section

Included with in the comments which contains author name, date of development & program details.

link (or) preprocessor section:

Provides preprocessor (or) like statements which direct the compiler to link functions from the sm library.

Definition section:

defines all symbolic constants refer to assigning a macro of a name to a constant.

The General Syntax of Symbolic constant is,

define constant_name constant_value

Global Declaration section:

It contains variable declarations which can be accessed anywhere with in the problem

ex: 5.2

```
int a, b;  
a = 5;  
b = 8;  
int c;
```

Function local

```
c = a + b;
```

local declaring

```
{ int d;  
  d = a - b;  
}
```

user define fun
add

```
{ int c;  
  d = a + b;  
}
```


Main Selection:

Divided into two portions:

1. Declaration part: → used to declare any variables in the main block of the program.

2. Executable part: → a set statements within

The open & close braces. The execution of the program begins at the opening braces and end at the closing braces.

User defined function (or) Sub program selection:

→ It is an main function.

→ Each user defined fun. contains the fun. name, arguments and return values

Program:

/* program to find the area of the circle */

#include <stdio.h>

#include <conio.h>

#define PI 3.14

void main()

{

float area, r;

clrscr();

printf("\n Enter the radius: \n");

scanf("%f", &r);

area = PI * (r * r);

printf("\n Area of the circle = %.2f", area);

getch()

}

/* executable part of the main fun () program */



O/p:

Enter the radius : 4.5
Area of the circle = 63.58

Program 2:

/* Program to find the sum of two number using
function */ → /* Documentation section */

#include <stdio.h> → /* Preprocessor section */

#include <conio.h> → /* Global variable declaration */

int a, b;

int add (int, int); → /* Fun. declaration */

Void main () → /* main of fun */

{ → /* local variable declaration */

int c;

clrscr();

printf ("In Enter the values of a & b : ");

scanf ("%d %d", &a, &b);

c = add (a, b);

printf ("In sum of %d + %d = %d", a, b, c);

getch();

/* execution part of the main() pgm */

int add (int a, int b) → /* subprogram of add () function
definition */

{ → /* local variable declaration */

c = a + b;

/* execution part of the function.

return c;

O/p:

Enter the value for a and b : 5 3

Sum of $5 + 3 = 8$.