NEW HORIZON COLLEGE

SUBJECT: PROBLEM SOLVING TECHNIQUES USING C

SEMESTER: I SEMESTER

COURSE: BCA

SUBJECT TEACHER: Dr. K.Chitra,
Assistant Professor,
Department of Computer Science

# Basic programing constructs

Chapter -2

- There are 3 basic programming constructs

  sequential

  selection

  Iteration

# Sequential Construct

In a sequential construct , program statements are executed one after another.

Input statement

Assignment statement

Output statement

# Selection construct

- It is also known as conditional construct. It is used to indicate decision in program.
- There are different kinds of selection constructs. They are

simple if

if…else

if…else if

nested if

Switch statement

# Iteration constructor

- Some of the statements have to be executed repeatedly, we can use repetition construct to perform many iterations.
- Conditional looping
- Unconditional looping

- Conditional looping

Many programs requires that a group of instructions be executed repeatedly until the logical condition has been satisfied.

(E-x) while , do..while

- Unconditional looping

Execution of a group of consecutive instructions is repeated some specified number of times.

(E-x ) for statement

# Chapter-3 overview of c

- C language is developed from BCPL (Basic combined Programming Language).
- It was developed by Dennis Ritchie in 1972.

# Importance of c

- It is robust.

**robustness** is the ability of a **computer** system to cope with errors during execution.

- C has the advantage of assembly language programming like bit manipulation and features of high level language programming like debugging.
- C is called as middle level language since it combines low level language and high level language.

- It is suited for writing system software and application software.
- It is platform independent and highly portable.
- C is a structured programing language. Program is divided into so many modules.
- C consists of variety of data types and powerful operators.
- C provides manipulation of internal processor registers.

# Applications of c

- Compilers
- Loaders
- Linkers
- Interpreters
- Operating system
- Database Management system
- Spread sheets

# Basic structure of C program

- Documentation section (comments)
- Preprocessor section (Header file)
- Definition section (Defining symbolic constants)
- Global Declaration section (global variables declared)
- Main()
- {
- Declaration part      (variable declaration)
- Executable part       (statements in c)
- }

- Subroutine section

  f1()

  f2()

  .

  .

  fn()

# Documentation section

- This section consists of set of comment lines starting with /* and ends with */
- This section helps the users to understand the program.
- This section is optional and are not executed by compiler.

E-x

/* To add two numbers */

# Preprocessor section

- This section contains header files which begins with a #symbol and are extended with .h
- This section is used to include the header file.

E-x

```
#include <stdio.h>
#include <math.h>
```

# Definition section

- Symbolic constants are included using #define e-x

```
#define pi 3.14
```

# Global Declaration section

- This section is used to define the variables that would be used in more than one function.
- This section should be declared before main() function.
- E-x

```
 int a;
main()
```

# main() function

- All c programs must contain main() function.
- It denotes the starting of the program.

# Braces

- Execution of the program begins at the opening brace { and ends at its closing brace.

# Declaration part

- This part is used to declare all the variables used in executable part.
- The syntax for declaration is

  datatype list of variables;

E-x

```
int a,b;
float b,d;
```

# Executable part

- This part of the program consists of a set of executable statements.
- Every statement in the declaration part and executable part ended with ;

e-x

```
 scanf(“%d%d”,&a,&b);
C=a+b;
printf(“%d”,c);
```

# Subroutine

- This section is optional and consists of all the user-defined functions.

E-x

```
Function1()
{
Printf(“hello world”);
}
Function2()
{
Printf(“welcome to the world”);
}
```

# Example program

```
  /* program to find circumference of the circle */
#include <stdio.h>
#define pi 3.14
main()
{
float circum,r;
Scanf(“%f”,&r);
circum=2*pi*r;
printf(“%f”,circum);
}
```

# Executing a c program

- Creation of program
- Compilation of a program
- Linking of a program
- Executing the program

# Creation of program

- Programs should be written in C editor.
- After typing the program ,save the program with extension .c

# Compilation of a program

- The source program should be compiled by using c compiler.
- If any syntax errors in a source program, then compiler checks for it.
- Once all the errors are corrected, the compiler converts the source program into object program.

# Linking the program

- All the library files are linked to the main program.

# Executing the program

- After compilation and linking, the executable object code will be loaded in the computer's main memory and the program is executed

# Programming style

- All the statements should be written in lower case letters.
- The programmer can write the statement anywhere between two braces.
- The opening and closing braces should be balanced.
- Use comment line whenever necessary.