Green University of Bangladesh

Lab Manual

CSE-104 (Structured Programming Lab)
Credit: 1.5, Contact hour: 3 hrs. Per week

Department of Computer Science and Engineering

Contents

Chapter 1

LAB-1

1.1 Introduction

C was initially used for system development work, in particular the programs that make up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as code written in assembly language. Some examples of the use of C might be:

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors
- Print Spoolers
- Network Drivers
- Modern Programs
- Databases
- Language Interpreters

- Utilities

All the C programs are writen into text files with extension ".c" for example hello.c. You can use "vi" editor to write your C program into a file.

A C program basically has the following form:

- Preprocessor Commands
- Functions
- Variables
- Statements & Expressions
- Comments

The following program is written in the C programming language. Open a text file **hello.c** using vi editor and put the following lines inside that file.

```c
#include <stdio.h>

int main()
{
   /* My first program */
   printf("Hello, World! \n");

   return 0;
}
```

Preprocessor Commands: These commands tells the compiler to do preprocessing before doing actual compilation. Like # include <stdio.h> is a preprocessor command which tells a C compiler to include stdio.h file before going to actual compilation.

Functions: are main building blocks of any C Program. Every C Program will have one or more functions and there is one mandatory function which is called main() function. This function is prefixed with keyword int which means this function returns an integer value when it exits. This integer value is retured using return statement.

The C Programming language provides a set of built-in functions. In the above example printf() is a C built-in function which is used to print anything on the screen.

Variables: are used to hold numbers, strings and complex data for manipulation.

Statements & Expressions: Expressions combine variables and constants to create new values. Statements are expressions, assignments, function calls, or control flow statements which make up C programs.

Comments: are used to give additional useful information inside a C Program. All the comments will be put inside /*...*/ as given in the example above. A comment can span through multiple lines.

1.1.1 Reserved Keywordss

The following names are reserved by the C language. Their meaning is already defined, and they cannot be re-defined to mean anything else.

auto	else	long	switch
break	enum	register	typedef
case	extern	return	union
char	float	short	unsigned
const	for	signed	void
continue	goto	sizeof	volatile
default	if	static	while
do	int	struct	_Packed
double			

While naming your functions and variables, other than these names, you can choose any names of reasonable length for variables, functions etc.

1.1.2 Datatypes

C has a concept of 'data types' which are used to define a variable before its use. The definition of a variable will assign storage for the variable and define the type of data that will be held in the location.

The value of a variable can be changed any time.

C has the following basic built-in datatypes.

- int
- float
- double
- char

There is not a boolean data type. C does not have the traditional view about logical comparison.

int - data type: int is used to define integer numbers.

```
{
    int Count;
    Count = 5;
}
```

float - data type: float is used to define floating point numbers.

```
{
    float Miles;
    Miles = 5.6;
}
```

double - data type: double is used to define BIG floating point numbers. It reserves twice the storage for the number. On PCs this is likely to be 8 bytes.

```
{
    double Atoms;
    Atoms = 2500000;
}
```

char - data type: char defines characters.

```
{
    char Letter;
    Letter = 'x';
}
```

1.1.3 Input and Output

- **Input:** In any programming language input means to feed some data into program. This can be given in the form of file or from command line. C programming language provides a set of built-in functions to read given input and feed it to the program as per requirement.

- **Output:** In any programming language output means to display some data on screen, printer or in any file. C programming language provides a set of built-in functions to output required data.

printf() function

This is one of the most frequently used functions in C for output. Try following program to understand printf() function.

```
#include <stdio.h>

main()
{
  int dec = 5;
  char str[] = "abc";
  char ch = 's';
  float pi = 3.14;

  printf("%d %s %f %c\n", dec, str, pi,  ch);
}
```

The output of the above would be:

```
5 abc 3.140000 c
```

Here %d is being used to print an integer, %s is being usedto print a string, %f is being used to print a float and %c is being used to print a character.

scanf() function

This is the function which can be used to to read an input from the command line. Try following program to understand scanf() function.

```c
#include <stdio.h>
int main()
{
    int testInteger;
    printf("Enter an integer: ");
    scanf("%d", &testInteger);
    printf("Number = %d",testInteger);
    return 0;
}
```

Here %d is being used to read an integer value and we are passing & test Integer to store the vale read input. Here & indicates the address of variable x.

This program will prompt you to enter a value. Whatever value you will enter at command prompt that will be output at the screen using printf() function. If you eneter a non-integer value then it will display an error message.

```
Enter an integer: 20
Read in 20
```

1.1.4 Exercises

Do the following tasks:

1. Write a C program to display "This is my first C Program".
2. Write a C Program to display your Id, Name, Dept. Name and University Name followed by a newline.
3. Write a C program to add two numbers (2 and 6) and display its sum.
4. Write a C program to multiply two numbers (4 and 5) and display its product.
5. Write a C program to add two numbers (5 and 8) and display its sum like (5 + 8 = 13).
6. Write a C program to input two numbers and display those numbers.
7. Write a C Program to input two numbers as input and display its sum.
8. Write a C Program to input two numbers as input and display its product.
9. Write a C Program to input two float numbers as input and display its sum [Follow the printing style of problem 5].

Chapter 2

LAB-2

2.1 Expressions

An expression is a formula in which operands are linked to each other by the use of operators to compute a value. An operand can be a function reference, a variable, an array element or a constant.

There are four types of expressions exist in C:

- Arithmetic expressions
- Relational expressions
- Logical expressions
- Conditional expressions

2.1.1 Example 1: Working of post-increment and pre-increment operator

Increment operators are used to increase the value by one while decrement works opposite increment. Decrement operator decrease the value by one.

Pre-increment (++i) - Before assigning the value to the variable, the value is incremented by one.

Post-increment (i++) - After assigning the value to the variable, the value is incremented.

The following is the syntax of pre and post increment.

$++variable\_name; //Pre-increment$

$variable\_name++; //Post-increment$

Code:

```
#include <stdio.h>

int main()
{
  int i;

  i = 4;

  printf( "%d\n", i );
  printf( "%d\n", i++ );   //post increment
  printf( "%d\n\n", i );

  i = 4;
  printf( "%d\n", i );
  printf( "%d\n", ++i );   //preincrement
  printf( "%d\n", i );

  return 0;
}
```

Output:

```
4
4
5

4
5
5
```

2.1.2 Example 2: Write C Program to solve simple Arithmetic Operators

Algorithm 1 Steps in pseudo code:

1: Step1: Start
2: step2: assign value in variable a, assign value in variable b and declare variable c
3: step3: Calculate c (c = a + b)
4: step4: Display result of addition, value of c on the monitor with the message "a+b = "
5: step5: Calculate c (c = a - b)
6: step6: Display result of subtraction, value of c on the monitor with the message "a-b = "
7: step7: Calculate c (c = a * b)
8: step8: Display result of multiplication, value of c on the monitor with the message "a*b = "
9: step9: Calculate c (c = a / b)
10: step10: Display result of division, value of c on the monitor with the message "a/b = "
11: step11: Calculate c (c = a % b)
12: step12: Display result of modulo-division , value of c on the monitor with the message "Remainder when a divided by b = "
13: step13: end

Code:

```c
#include <stdio.h>
int main()
{
    int a = 9,b = 4, c;

    c = a+b;
    printf("a+b = %d \n",c);
    c = a-b;
    printf("a-b = %d \n",c);
    c = a*b;
    printf("a*b = %d \n",c);
    c = a/b;
    printf("a/b = %d \n",c);
    c = a%b;
    printf("Remainder when a divided by b = %d \n",c);

    return 0;
}
```

Output:

```
a+b = 13
a-b = 5
a*b = 36
a/b = 2
Remainder when a divided by b=1
```

2.1.3 Example 3: Write C Program to find the Volume of a Cylinder(take input from user).

Algorithm 2 Steps in pseudo code:

1: Step1: Start
2: step2: declare float variable vol, r and h
3: step3: Display a message on the monitor "enter radius: "
4: step4: read r
5: step5: Display a message on the monitor "enter height: "
6: step6: read h
7: step7: Calculate vol (vol =(22*r*r*h)/7)
8: step8: Display result of Volume of a Cylinder, value of vol on the monitor with the message "VOC: "
9: step9: end

Code:

```c
#include<stdio.h>
int main()
{

    float vol,r,h;
    printf("enter radius: ");
    scanf("%f",&r);
    printf("enter height: ");
    scanf("%f",&h);

    vol=(22*r*r*h)/7;
    printf("VOC: %f\n",vol);
    return 0;
}
```

Output:

```
enter radius: 7
enter height: 7
VOC: 1078.000000
```

2.1.4 Exercises

Do the following tasks:

1. Write a C program to enter two numbers and perform all arithmetic operations.
2. Write a C Program to Calculate Area and Circumference of Circle.
3. Write a C program to enter length in centimeter and convert it into meter and kilometer.
4. Write a C Program to Calculate Area of a Rectangle, take height and width as user input.
5. Write a C Program to Calculate Area of a Square, take length of one side as user input.
6. Write a C program to enter temperature in °Celsius and convert it into °Fahrenheit.
7. Write a C program to enter temperature in Fahrenheit(°F) and convert it into Celsius(°C).
8. Write a C program to enter marks of five subjects and calculate total and average marks.

Chapter 3

LAB-3

3.1 Selection Statement

The if-else statement in C is used to perform the operations based on some specific condition which is called selection statement. The operations specified in if block are executed if and only if the given condition is true.

There are the following variants of if statement in C language.

- If statement
- If-else statement
- If else-if ladder
- Nested if

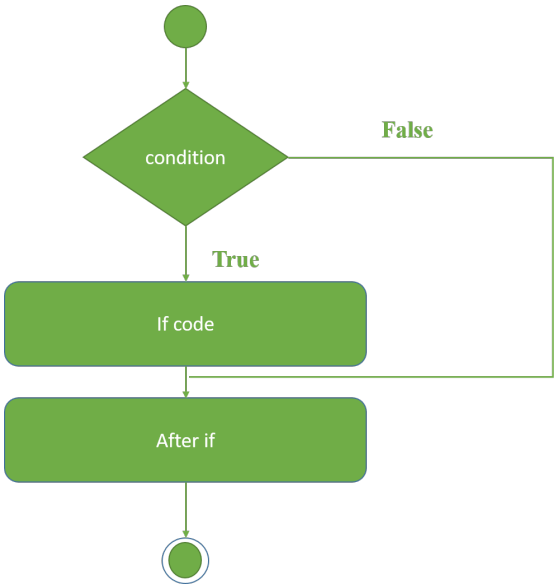
3.1.1 If Statement

The if statement is used to check some given condition and perform some operations depending upon the correctness of that condition.

It is mostly used in the scenario where we need to perform the different operations for the different conditions. The syntax of the if statement is given below.

```
if(expression){
//code to be executed
}
```

Flowchart of if statement:



3.1.2 Example 1: Write C Program to check If statement.

Algorithm 3 Steps in pseudo code:

1: Step1: Start
2: step2: assign value in variable i
3: step3: The if statement evaluates the i expression inside the parenthesis ().
4: step4: If the i expression is evaluated to true, statements inside the body of if are executed. or, If the i expression is evaluated to false, statements inside the body of if are not executed.
5: step5: The statements after the body of if are executed.
6: step6: end

Code:

```c
#include <stdio.h>

int main() {
    int i = 10;

    if (i > 20)
    {
       printf("10 is less than 20");
    }

    printf("I am Not in if");
    return 0;
}
```

Output:

```
After if executed.

Process returned 0 (0x0)   execution time : 0.028 s
Press any key to continue.
```

3.1.3 Example 2: Write a C program to check the largest number(take three number from user).

Code:

```c
#include <stdio.h>
int main()
{
    int a, b, c;
     printf("Enter three numbers: \n");
    scanf("%d %d %d",&a,&b,&c);
    if(a>b && a>c)
    {
        printf("%d is largest. \n",a);
    }
    if(b>a  && b > c)
    {
        printf("%d is largest. \n",b);
    }
    if(c>a && c>b)
    {
        printf("%d is largest. \n",c);
    }
    if(a == b && a == c)
    {
        printf("All are equal. \n");
    }
    return 0;
}
```

Output:

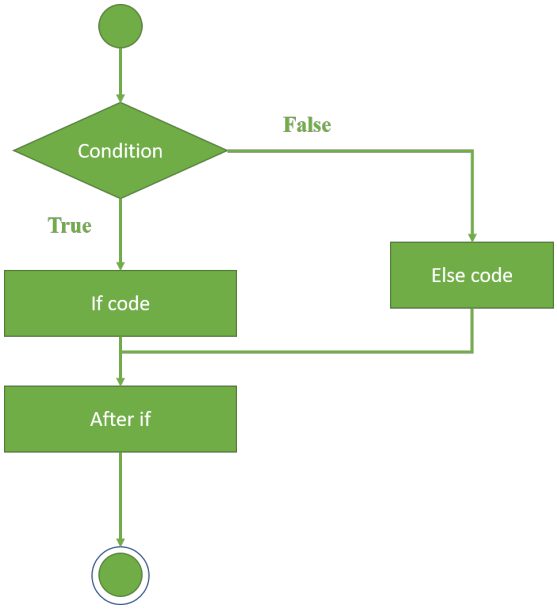
```
Enter three numbers:
45
97
15
97 is largest.
```

3.1.4 If-else Statement

The if-else statement is used to perform two operations for a single condition. The if-else statement is an extension to the if statement using which, we can perform two different operations, i.e., one is for the correctness of that condition, and the other is for the incorrectness of the condition. Here, we must notice that if and else block cannot be executed simultaneously. Using if-else statement is always preferable since it always invokes an otherwise case with every if condition. The syntax of the if-else statement is given below.

```
if(expression){
//code to be executed if condition is true
}else{
//code to be executed if condition is false
}
```

Flowchart of if-else statement:



3.1.5 Example 1: Write a C program to Check whether an integer is odd or even(take input from user).

Algorithm 4 Steps in pseudo code:

1: Step1: Start
2: Step2: Declare variable number
3: Step3: read number
4: Step4: If the remainder of that number is 0, it is evaluated to true. The statements inside the body of if are executed. & The statements inside the body of else are skipped from execution.
5: Step5: If the remainder of that number is not 0, it is evaluated to false. The statements inside the body of else are executed. & The statements inside the body of if are skipped from execution.
6: Step6: End

Code:

```c
#include <stdio.h>
int main() {
    int number;
    printf("Enter an integer: ");
    scanf("%d", &number);

    // True if the remainder is 0
    if  (number%2 == 0) {
        printf("%d is an even integer.",number);
    }
    else {
        printf("%d is an odd integer.",number);
    }

    return 0;
}
```

Output:

```
Enter an integer: 7
7 is an odd integer.
```

3.1.6 Example 2: Write a C program to check whether a person is eligible to vote or not.(take input from user and eligible age to vote is 18 or greater than 18).

Code:

```c
#include <stdio.h>
int main()
{
   int age;
   printf("Enter your age?");
   scanf("%d",&age);
   if(age>=18)
   {
      printf("You are eligible to vote...");
   }
   else
   {
      printf("Sorry ... you can't vote");
   }
}
```

Output:

```
Enter your age?18
You are eligible to vote...
Enter your age?13
Sorry ... you can't vote
```

3.1.7 Exercises

Do the following tasks:

1. Write a C program to check whether a number is negative, positive or zero.
2. Write a C program to check whether a number is divisible by 5 and 11 or not.
3. Write a C program to check whether a number is even or odd.
4. Write a C program to check whether a year is leap year or not.
5. Write a C program to input week number and print week day.
6. Write a C program to input angles of a triangle and check whether triangle is valid or not.
7. Write a C program to input all sides of a triangle and check whether triangle is valid or not.
8. Write a C program to check whether the triangle is equilateral, isosceles or scalene triangle.