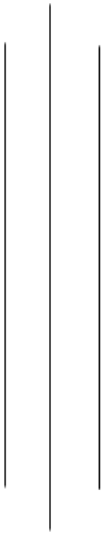
INSTITUTE OF ENGINEERING

Pulchowk Campus, Lalitpur



Subject: C Programming

Lab Report 1 and 2

Tittle: **Introduction and Variables**

Submitted by:

Susheel Thapa 077BCT090

Submitted to:

Department of Electronics and

Computer Engineering

Checked by

Content of Lab Report:

Background Information

- C Programming
- Editor Used
- Compiler
- Data Types
- Variables and its types

Code and Output (Analysis)

- Source Code
- Output
- Analysis

Conclusion

Background Information

What is C Programming?

C programming is a general-purpose, procedural, imperative computer programming language developed in 1972 by Dennis M. Ritchie at the Bell Telephone Laboratories to develop the UNIX operating system. C is the most widely used computer language.

Why to Learn C Programming?

- Easy to learn
- Structured language
- It produces efficient programs
- It can handle low-level activities
- It can be compiled on a variety of computer platforms

Editor

Here, I have used Visual Studio Code as my editor. You can download the editor from Download Visual Studio Code - Mac, Linux, Windows . Select your operating system and download it.

Compiler

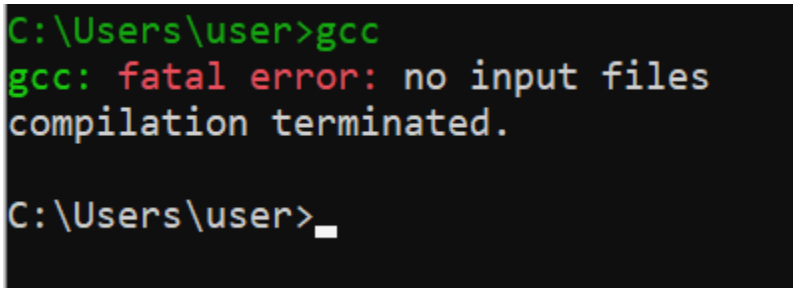
Here, I have used **gcc** as my compiler provided by MinGWw64. You can download it via Download MinGW-w64 - for 32 and 64 bit Windows from SourceForge.net. Your download will start automatically. Run the downloaded .exe file. After, you have install MinGW-w64, you need to configure it.

1. In the Windows search bar, type 'settings' to open your Windows Settings.
2. Search for Edit environment variables for your account.
3. Choose the Path variable and then select Edit.
4. Select New and add the Mingw-w64 destination folder path to the system path. The exact path depends on which version of Mingw-w64 you have installed and where you installed it. If you used the settings above to install Mingw-w64, then add this to the path: **C:\Program Files\mingw-w64\x86_64-8.1.0-posix-seh-rt_v6-rev0\mingw64\bin.**
5. Select OK to save the updated PATH. You will need to reopen any console windows for the new PATH location to be available.

Check your installation

Open command prompt or powershell and type:

```
C:\Users\user>gcc --version
gcc (x86_64-posix-seh-rev0, Built by MinGW-W64 project) 8.1.0
Copyright (C) 2018 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```



If you get similar result, you are good to go.

Basic Structure of C programming

A C Program starts with a main function and execute instruction present inside it.

Each instruction is terminated with semi colon (;).

Rules

- Program execution start from main ()
- All statements terminate with semicolon (:)
- Instructions are case-sensitive.
- Instructions are executed in the same order in which they are written.

Header Files

#include<stdio.h>

#include the pre-processor expands the line in turn and includes a copy of the standard header file stdio.h in the code as well. We have included stdio.h because we are using print () (used to print text or value in c program).

Main Function

Syntax:

int main () {

code

return 0;

}

From this function the execution of code starts. All the statement, declaration, function call, are written here.

Comments in C

Comments are those pieces of lines that for neglected by compiler during execution. Comments helps us to understand other code.

- **Single Line Comment**

  //This is single line comment. It starts with //.

- **Multi Line Comment**

  /*

  This is

  Multi Line

  Comments

  */

Data Types In C

In C, we have 3 data types:

1. int
2. float
3. char

Int refer to integer. It is used to store the integer number. For examples: 1, 2, 6, 40, etc
Float refer to floating point number. It is used to store the floating-point number (Number with decimal value). For examples: 1.23, 1.56, 6.25, etc.

Char refers to character. It is used to store character. For examples: a, u, w, %, &, etc.

Variables in C and its Types

Variable are the container which stores the value. Just like we have bottle to store water, container to store rice, daal, etc. In the same manner, we have different container in C to store different items.

A = 56

b = 4.7

c = 'A'

Before assigning value to a variable, we must declare its data types:

int A

float b

char c

Rules for Naming the Variables

- First character must be an alphabet or underscore (-)
- No commas, blanks are allowed in variable name
- No special symbol other than (-) is allowed
- Variable name is case sensitive.

  i.e., Name and name are different

Getting start with C

```
#include <stdio.h>

int main(void)

{

printf("Hello C Program");

return 0;

}
```

Dissection of Program

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

Lines begin with # is used to communicate with the preprocessor. This #include line causes the prepocessor to include a copy of the header files studio.h at this point of code. The angular bracket<> indicated that it is to be found in usual place, which is system-dependent. We have included this file because it contains the information about the printf() function.

```
int main(void)
```

This is first line of function definition of main ().(We write parentheses after the name main to remind the reader that main () is a function.) The words int and void are keywords, also called reserved words which have special meaning in C. There are 32 reserved words in C.

```
int main(void)

{…….
```

Every program has function main (). Program execution always starts with this function. The top line should be read as **"main () is a function that takes no arguments and returns an int value."** Here, the keyword int tells the compiler that this function returns a value of type int. The parentheses following main indicate to the compiler that main is a function. The keyword void indicates to the compiler that this function takes no arguments.

{

Braces surround the body of a definition. They are used to group statements together.

printf()

The C system contains a standard library of functions that can be used in programs. This is a function from the library that prints on the screen. We included the header file stdio.h because it provides certain information to the compiler about the function printf().

"Hello C Program"

A string constant in C is a series of characters surrounded by double quotes. This string is an argument to the function printf (), and it controls what gets printed.

printf("Hello C Program");

This is a call to the printf() function. In a program, the name of a function followed by parentheses causes the function to be called. It prints its argument, a string constant, on the screen.

return 0;

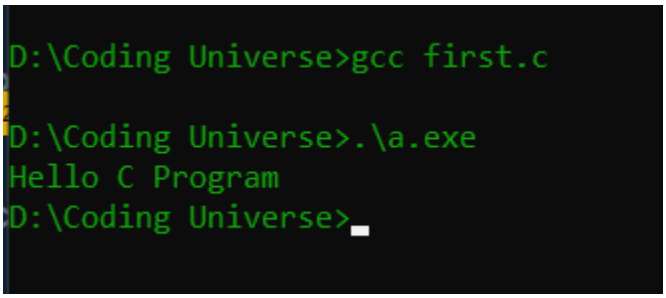
This is a return statement. It causes the value zero to be returned to the operating system. Our use of this return statement keeps the compiler happy. If we do not use it, the compiler will complain.

}

The right brace matches the left brace above, ending the function definition for main ().

Running the program

- Save it as first.c and open command prompt on the folder where you have the file
- Run gcc first.c
- Then .\a.exe



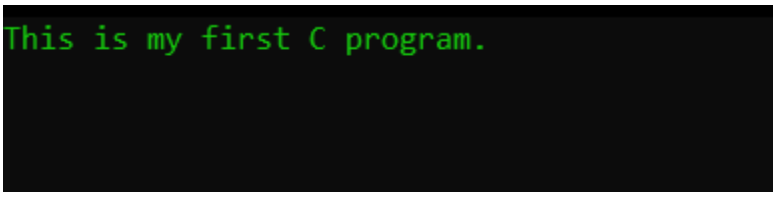
Lab 1

1.Type the following program and see the output

Source Code:

```
#include <stdio.h>
#include <conio.h>
void main(void)
{
   printf("This is my first C program.");
   getch();
}
```

Output:

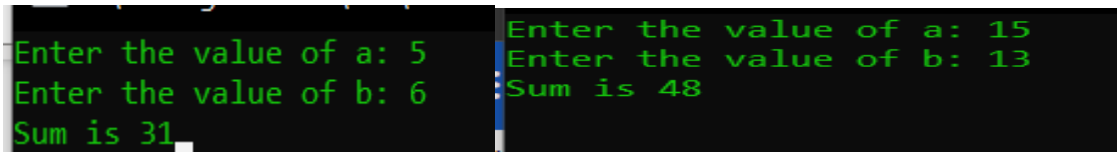


2.Type the following program and run with different input.

Source Code:

```
#include <stdio.h>
#include <conio.h>
int main (){
    int s, a, b, c = 20;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("Enter the value of b: ");
    scanf("%d", &b);
    s = a + b + c;
    printf("Sum is %d", s);
    getch();
    return 0;
}
```

Output:

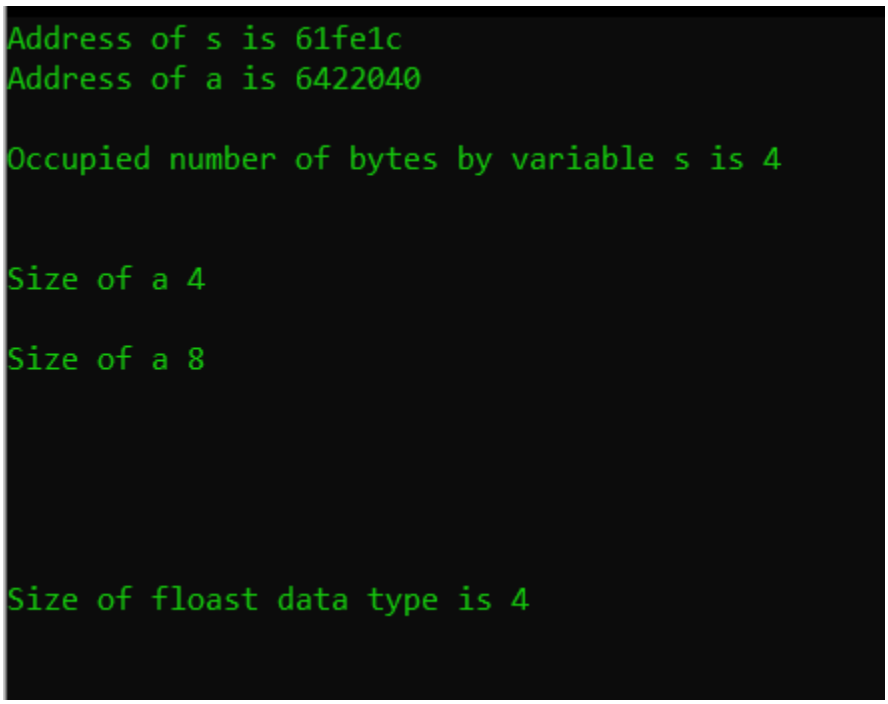


3.Type the following program and run and see the output.

Source Code:

```
#include <stdio.h>
#include <conio.h>
void main ()
{
   int s, a, b;
   float p;
   system("cls");
   printf("Address of s is %x\n", &s);
   printf("Address of a is %x\n\n", &a);
   printf("Occupied number of bytes by variable s is %d\n\n\n", sizeof(s));
   printf("Size of a %d\n\n", sizeof(p));
   printf("Size of a %d\n\n\n\n\n\n", sizeof(1.5));
   printf("Size of floast data type is %d", sizeof(float));
   getch();
}
```

Output:


Address of s is 61fe1c
Address of a is 6422040
Occupied number of bytes by variable s is 4
Size of a 4
Size of a 8
Size of floast data type is 4

4.Write a program to calculate the area, circumference of a circle of radius r.

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include <stdlib.h>
void main ()
{float area, circumference, r;
   system("cls")
   printf("Radius of circle:");
   scanf("%f", &r);
   area = M_PI * pow (r, 2);
   circumference = 2 * M_PI * r;
   printf("\nArea of Circle: %.2f m^2\n", area);
   printf("\nCircumference of circle: %.2f m.", circumference);
   getch();
}
```

Output:

```
Radius of circle:5

Area of Circle: 78.54 m^2

Circumference of circle: 31.42 m.
```

5.Write a program to calculate the volume of sphere of radius r.

Source code:

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include<stdlib.h>
void main ()
{
   float volume, radius;
   system("cls");
   printf("Radius of Sphere: ");
   scanf("%f", &radius);
   volume = (float)4 / 3 * M_PI * pow (radius, 3);
   printf("\n\nThe volume of the sphere is: %.2f m^3", volume);
   getch();
}
```

Output:

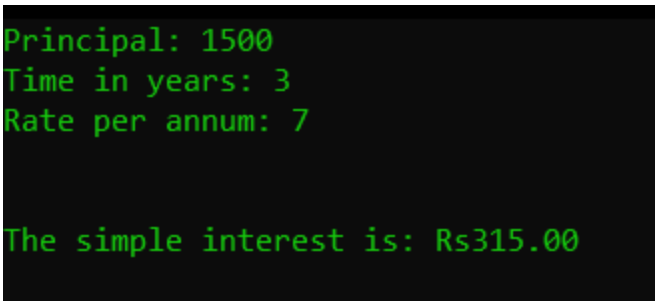
```
Radius of Sphere: 56

The volume of the sphere is:735618.63 m^3
```

6.Write a program to calculate the simple interest. Read values of P, T, R

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main ()
{
   float principal, time, rate, simple_interest;
   system("cls");
   printf("Principal: ");
   scanf("%f", &principal);
   printf("Time in years: ");
   scanf("%f", &time);
   printf("Rate per annum: ");
   scanf("%f", &rate);
   simple_interest = (principal * time * rate) / 100;
   printf("\n\nThe simple interest is: Rs%.2f", simple_interest);
   getch();
}
```

Output:

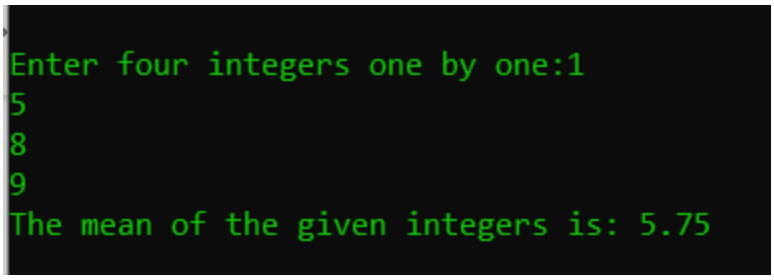


7.Write a program to read values of x and y from the user and evaluate the expression v=x^3+y^2-100/x.

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include <stdlib.h>
void main ()
{
   float x, y, expression_value;
   system("cls");
   printf("Enter the value of x and y respectively:");
   scanf("%f%f", &x, &y);
   expression_value = pow (x, 3) * pow (y, 2) - (float)100 / x;
   printf("Expression:\n v=(x^3+y^2) -(100/x)");
   printf("\n\nThe value of the expression is: %.2f", expression_value);
   getch();
}
```

Output:

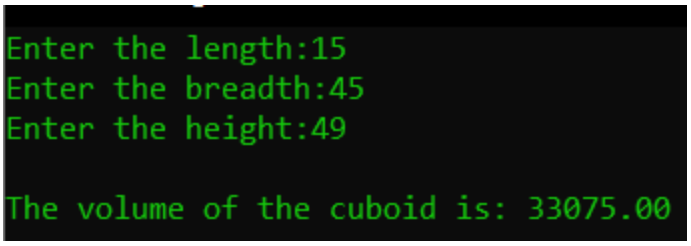


8.Write a program to read four integers from the user and display mean of the numbers

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main ()
{
   int a, b, c, d;
   float mean;
   system("cls");
   printf("\nEnter four integers one by one:");
   scanf("%d%d%d%d", &a, &b, &c, &d);
   mean = (float) (a + b + c + d) / 4;
   printf("The mean of the given integers is: %.2f", mean);
   getch();
}
```

Output:



9.Write a program to read l, b and h of a cuboid and display its volume.

Source code:

```
#include <stdio.h>
#include <conio.h>
#include<stdlib.h>
void main ()
{
   float l, b, h, volume;
   system("cls");
   printf("Enter the length:");
   scanf("%f", &l);
   printf("Enter the breadth:");
   scanf("%f", &b);
   printf("Enter the height:");
   scanf("%f", &h);
   volume = (l * b * h);
   printf("\nThe volume of the cuboid is: %.2f", volume);
   getch();
}
```

Output:

```
Enter the price of two pen: 15
Enter the price of five copies: 50
The total price after discount is: Rs58.50
```

10. Write a program to read price of two pens and five copies of same type and calculate the price after discounting 10%

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main ()
{
   float pen_price, copies_price, total_price, discount;
   system("cls");
   printf("Enter the price of two pen: ");
   scanf("%f", &pen_price);
   printf("Enter the price of five copies: ");
   scanf("%f", &copies_price);
   discount = 0.1;
   total_price = (pen_price + copies_price) * (1 - discount);
   printf("The total price after discount is: Rs%.2f", total_price);
   getch();
}
```

Output:

```
Enter the price of two pen: 56
Enter the price of five copies: 12
The total price after discount is: Rs61.20
```

11.Write a program to read time given for C programming study a day at your home in hours, minutes and seconds and display the total time in seconds in 30 days.

Source code:

```
#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

void main()

{

    float h, m, s, month;

    system("cls");

    printf("\nEnter time you study C programming per day in hours minutes and seconds:");

    scanf("%f%f%f", &h, &m, &s);

    month = (h * 3600 + m * 60 + s) * 30;

    printf("\n\nThe seconds studied for C programming in a month is: %.3f seconds", month);

    getch();

}
```

Output:

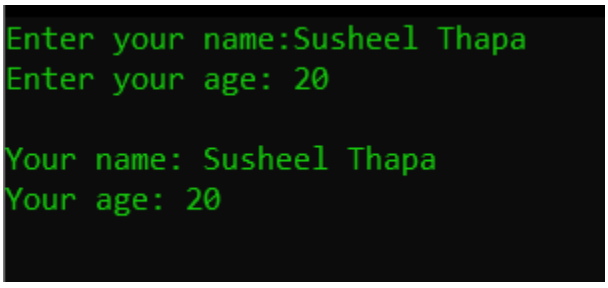
```
Enter time you study C programming per day in hours minutes and seconds:4
15
15


The seconds studied for C programming in a month is: 459450.000 seconds
```

12.Write a program to read name, age of a person, and display them:

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include<stdlib.h>
void main ()
{
   char name[40];
   int age;
   system("cls");
   printf("Enter your name:");
   gets(name);
   printf("Enter your age: ");
   scanf("%d", &age);
   printf("\nYour name: %s", name);
   printf("\nYour age: %d", age);
   getch();
}
```

Output:

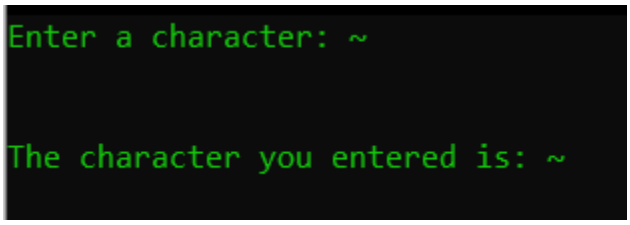


13. Write a program to read a character and display it.

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include<stdlib.h>
void main ()
{
   char x;
   system("cls");
   printf("Enter a character: ");
   scanf("%c", &x);
   printf("\n\nThe character you entered is: %c", x);
   getch();
}
```

Output:



Lab 2

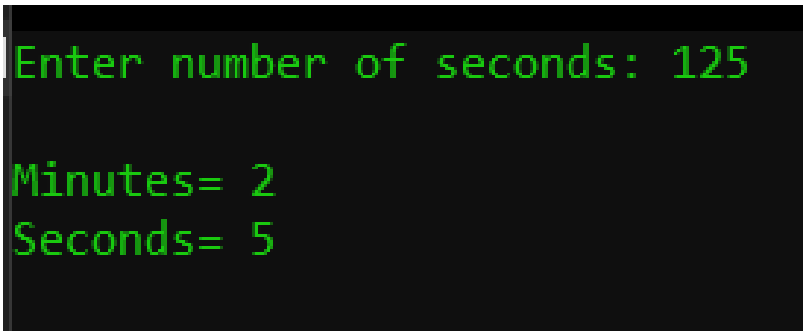
1.Program that inputs seconds as input and convert to minutes.

Source Code:

```
#include <stdio.h>
#include <conio.h>
void main(void)
{
```

```
int seconds, min;
    system("cls");
    printf("Enter number of seconds: ");
scanf("%d", &seconds);
    min = seconds / 60; /*Integer Division*/
seconds = seconds % 60; /*Integer Division*/
    printf("\nMinutes= %d", min);
printf("\nSeconds= %d", seconds);
printf("Press any key to exit...");
getch();
}
```

Output:



2.A program to illustrate prefix increment operator

Source Code:

```
#include <stdio.h>
#include <conio.h>
void main ()
{
int x = 5, v;
```

```
system("cls");
v = ++x * ++x + ++x;
printf("v = %d, x = %d", v, x);
printf("Press any key to exit...");
getch();
}
```

Output

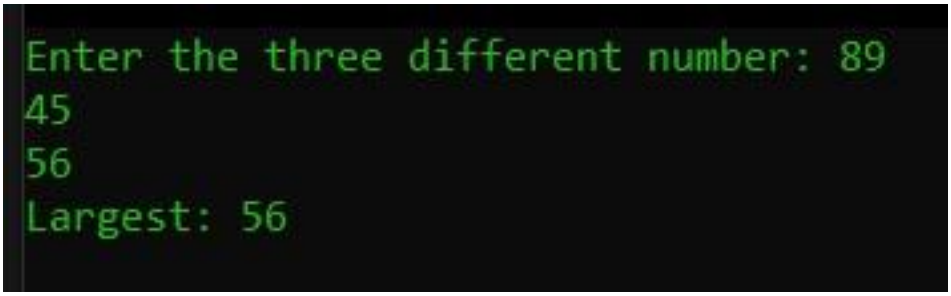
3.Program to read the three different integer from the user and display the largest among number them

Source code:

```
#include <stdio.h>
#include <conio.h>
void main () {
    int a, b, c, l;
    system("cls");
    printf("Enter the three different number: ");
    scanf("%d%d%d", &a, &b, &c);
    if (a > b && b > c) {
    l = a;
    }
    else if (b > c) {
    l = b;
    }
    else {
```

```
        l = c;
        }
        printf("Largest: %d", l);
        printf("Press any key to exit...");
        getch();
}
```

Output:



4.Type, Compile, run and observe and think about the output of the following program

Source code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main ()
{
        int x, y, z;
        system("cls");
        x = 30000, y = 20000;
```

```
        z = x + y;
        printf("Sum = %d", z);
        printf("Press any key to exit...");
        getch();
    }
```

Output:

5. Type, run and observe the output of the following program

Source code:

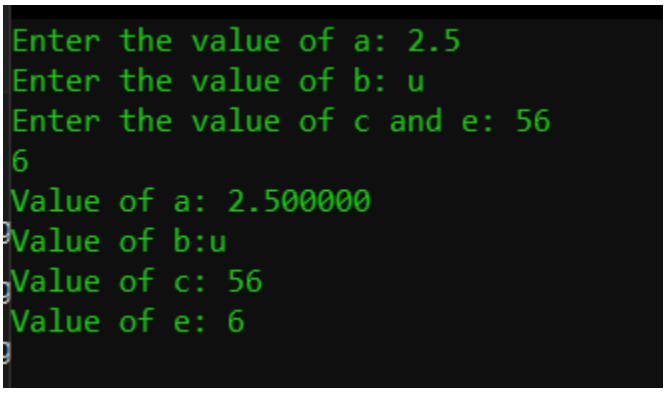
```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int main ()
{
float a;
char b;
long int c;
    unsigned int e;
    system("cls");
     printf("Enter the value of a: ");
 scanf("%f", &a);
printf("Enter the value of b: ");
scanf(" %c", &b);
```

```
printf("Enter the value of c and e: ");
        scanf("%ld%u", &c, &e);
                printf("Value of a: %f\nValue of b: %c\nValue of c: %ld\nValue of e: %u",
                a, b, c, e);
                 getch();
 return 0;
}
```

No table of contents entries found.

No table of contents entries found.

Output:



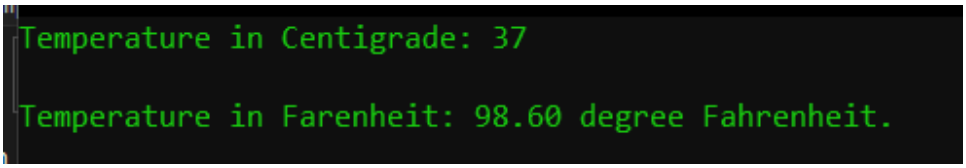
6.Write a program to convert the given centigrade measure into Farenheit using relation

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int main ()
```

```
{
float fahrenheit, celsius;
system("cls");
printf("Temperature in Centigrade: ");
scanf("%f", &celsius);
fahrenheit = 1.8 * celsius + 32;
        printf("\nTemperature in Farenheit: %.2f degree Fahrenheit. ",
        fahrenheit);
getch();
return 0;
}
```

Output:



7.Write a program to compute equivalent resistance of two resistor R1 and R2 when they are in series and in parallel

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int main ()
{
float resistance_one, resistance_two, resistance_series, resistance_parallel;
system("cls");
```

printf("Value of Resistance One: ");

scanf("%f", &resistance_one);

printf("Value of Resistance Two: ");

scanf("%f", &resistance_two);

resistance_series = resistance_one + resistance_two;

*resistance_parallel = (resistance_one * resistance_two) / (resistance_one + resistance_two);*

printf("\nResistence in Series: %f\n", resistance_series);

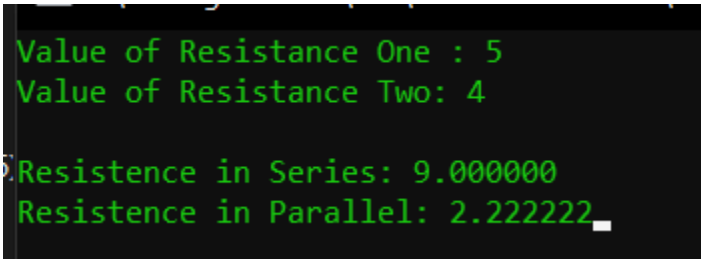
printf("Resistence in Parallel: %f", resistance_parallel);

getch();

return 0;

}

Output:



8.Write a program to read two end points of a line, compute their mid point and display it.

Source code

#include <stdio.h>

#include <conio.h>

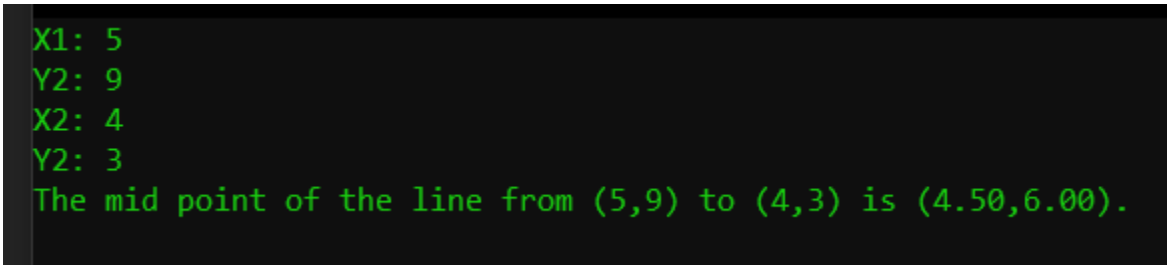
#include <stdlib.h>

int main ()

{

```
int x1, y1, x2, y2;
float mid_point_x, mid_point_y;
system("cls");
printf("X1: ");
scanf("%d", &x1);
printf("Y2: ");
scanf("%d", &y1);
printf("X2: ");
scanf("%d", &x2);
printf("Y2: ");
scanf("%d", &y2);
mid_point_x = (float)(x1 + x2) / 2;
 mid_point_y = (float)(y1 + y2) / 2;
printf("The mid point of the line from (%d,%d) to (%d,%d) is (%.2f,%.2f).", x1, y1,
        x2, y2, mid_point_x, mid_point_y);
        printf("Press any key to exit...");
getch();
return 0;
}
```

Output:



9.Write a program to read number of girls and boys in your class and display their ratio of girls to boys.

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int main ()
{
int girls_number, boys_number;
float ratio;
system("cls");
printf("Enter the number of girls: ");
scanf("%d", &girls_number);
printf("Enter the number of boys: ");
scanf("%d", &boys_number);
system("cls");
ratio = ((float)girls_number) / ((float)boys_number);
printf("No. of Girls: %d\n", girls_number);
 printf("No. of Boys: %d\n", boys_number);
printf("\nRatio: %.2f #Girls to Boys\n", ratio);
printf("Press any key to exit...");
getch();
return 0;
}
```

Output:

```
Enter the number of girls: 15
Enter the number of boys: 26
```

```
No. of Girls : 15
No. of Boys : 26

Ratio: 0.58   #Girls to Boys
```

10.Write a program to evaluate the following expression

S =x^5 + 0.2*x*y + y^7

L = (a+b)^((2x+y)/(p-q)) +c-100

r = A/B [Where A and B are integers]

R = ((u/x+v/y) ^5)/((((p^2)/(3u^2.5)) -(q/2v)) ^3.5)

Source Code:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <math.h>
int main ()
{
   int x, y, a, b, p, q, c, A, B, u, v;
   float S, L, r, R;
   system("cls");
```

```
printf("Expression 1: \n\n\t S =x^5 + 0.2*x*y + y^7\n");
printf("\nValue of x: ");
scanf("%d", &x);
printf("Value of y: ");
scanf("%d", &y);
S = pow (x, 5) + (float)0.2 * x * y + pow (y, 7);
printf("\nValue of expression S =x^5 + 0.2*x*y + y^7 is %.3f ", S);

printf("\n\nExpression 2: \n\n\t L = (a+b)^((2x+y)/(p-q)) +c-100\n");
printf("\nValue of x: ");
scanf("%d", &x);
printf("Value of y: ");
scanf("%d", &y);
printf("Value of a: ");
scanf("%d", &a);
printf("Value of b: ");
scanf("%d", &b);
printf("Value of p: ");
scanf("%d", &p);
printf("Value of q: ");
scanf("%d", &q);

float base2 = (float) (a + b);
float power2 = (float)(2 * x + y) / (p - q);
```

```
    L = pow (base2, power2) + c - 100;
    printf("\nValue of expression L = (a+b)^((2x+y)/(p-q)) +c-100 is %.3f ", L);


    printf("\n\nExpression 3: \n\n\t r = A/B [Where A and B are integers] \n");
    printf("\nValue of A: ");
    scanf("%d", &A);
    printf("Value of B: ");
    scanf("%d", &B);
    r = A / B;
    printf("\nValue of expression r = A/B [Where A and B are integers] without
        typecasting is %.2f ", r);
    r = (float)A / B;
    printf("\nValue of expression r = A/B [Where A and B are integers] with typecasting is
%.2f ", r);


    printf("\n\nExpression 4: \n\n\t R = ((u/x+v/y) ^5)/((((p^2)/(3u^2.5)) -(q/2v)) ^3.5)
\n");
    printf("\nValue of u: ");
    scanf("%d", &u);
    printf("Value of v: ");
    scanf("%d", &v);
    printf("Value of x: ");
    scanf("%d", &x);
    printf("Value of y: ");
    scanf("%d", &y);
    printf("Value of p: ");
```

```
    scanf("%d", &p);
    printf("Value of q: ");
    scanf("%d", &q);
    float numerator_base_4 = ((float)u / x + v / y);
    float numerator = pow (numerator_base_4, 5);
    float denominator_first_part = ((float)p * p) / (3 * pow (u, 2.5));
    float denominator_second_part = (float)q / (2 * v);
    float denominator_base = (denominator_first_part - denominator_second_part);
    float denominator = pow(denominator_base, 3.5);
    R = numerator / denominator;

    printf("\nValue of expression R = ((u/x+v/y) ^5)/((((p^2)/(3u^2.5)) -(q/2v)) ^3.5) is
%.3f ", R);

    getch();
    return 0;
}
```

Output:

```
Expression 1:

        S =x^5 + 0.2*x*y + y^7

Value of x: 1
Value of y: 1

Value of expression S =x^5 + 0.2*x*y + y^7 is 2.200

Expression 2:

        L = (a+b)^((2x+y)/(p-q))+c-100

Value of x: 1
Value of y: 1
Value of a: 1
Value of b: 1
Value of p: 1
Value of q: 0

Value of expression L = (a+b)^((2x+y)/(p-q))+c-100 is -84.000

Expression 3:

         r = A/B [Where A and B are integers]

Value of A: 15
Value of B: 37

Value of expression r = A/B [Where A and B are integers] without typecasting is 0.00
Value of expression r = A/B [Where A and B are integers] with typecasting is 0.41

Expression 4:

         R = ((u/x+v/y)^5)/((((p^2)/(3u^2.5))-(q/2v))^3.5)

Value of u: 1
Value of v: 2
Value of x: 1
Value of y: 1
Value of p: 1
Value of q: 1

Value of expression R = ((u/x+v/y)^5)/((((p^2)/(3u^2.5))-(q/2v))^3.5) is 1454589.500000
```

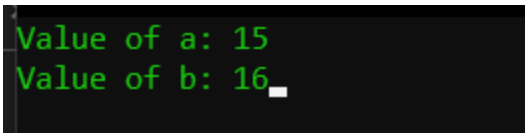
11.Write a program to swap value of two variable

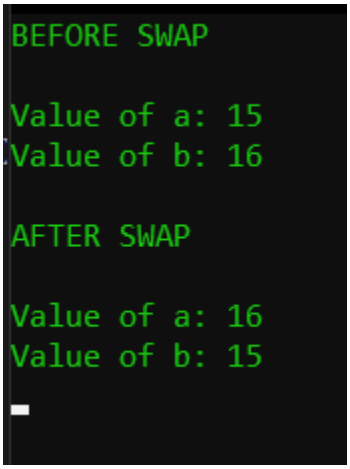
Source Code:

#include <stdio.h>

```
#include <conio.h>
#include <stdlib.h>
int main ()
{
int a, b, c;
system("cls");
printf("Value of a: ");
scanf("%d", &a);
printf("Value of b: ");
scanf("%d", &b);
system("cls");
printf("BEFORE SWAP\n\n");
printf("Value of a: %d \n", a);
printf("Value of b: %d \n", b);
c = a;
a = b;
b = c;
 printf("\nAFTER SWAP\n\n");
printf("Value of a: %d \n", a);
printf("Value of b: %d \n", b);
printf("Press any key to exit...");
 getch();
return 0;
}
```

Output:





Analysis and Discussion

In our first lab report here we have written a lot of programs and run it many times. Through this, we are able to know the respective syntax of the element like printf(), scanf(), gets () and many more.

Moreover, it provides us brief information about work flow of compiler, variable, data type, address, format specifier escape character, inbuilt keywords and many more.

Conclusion

Therefore the objective of the first lab session was completed with successful completion of the given exercise questions and better understanding of the aspects of the C language data types