

**Birla Institute of Technology & Science, Pilani, K. K. Birla Goa Campus**  
**Computer Programming (CS F111)**  
**Second Semester 2017-2018**  
**Lab-5 (Introduction to C)**

---

**Objectives**

1. Getting started with C.
  2. Compiling and executing your program.
  3. Options for controlling the kind of output.
  4. To understand conditional constructs in C.
    - a. **if, if else** with a calculator program example
    - b. **switch** with a calculator program example
  5. Exercises
- 

## 1. Getting Started with C

Communicating with a computer requires speaking in the language that computer can understand, which immediately rules out English as the language of communication with computer. C is a programming language which we will be using for it. It was developed at AT&T's Bell Laboratories of USA in 1972 and was designed and written by a man named Dennis Ritchie.

Armed with the knowledge about the type of variables and keywords, let's try our first C program. Each instruction in a C program is written as a separate statement. These statements must appear in the same order in which we wish them to be executed; unless the logic of the problem demands transfer of control to a statement which is out of sequence.

## 2. Your First C Program

The following C program calculates the simple interest.

`/*Program to calculate the simple interest*/`. This is how a comment is written. Comments are just to improve the readability of the program. Make a habit of writing comments from the very beginning.

```
#include<stdio.h>      /* including header file */
int main(void)
{
    int pri, time;      /*variable declarations*/
    float rate, si;
    pri = 2000;         /* Initializations */
    time = 4;
    rate = 9.5;
    si = pri * time * rate / 100; /* assignment statement */
    printf ("The Calculated Simple Interest is %f", si);
    return (0);
}
```

Where to write this program? You need a text editor for this. Recall that in UNIX, we have a text editor, called vi, and you are already armed with the basic knowledge of it. So, to write the program, type: vi *filename.c*:

```
[CPSEC1@localhost ~]$ vi sinterest.c
```

Remember, all C programs files has an extension *.c*. After typing, save the file and come to the prompt.

Now we're going to need a *compiler*. What is a compiler? It is a program that turns the C code into an *executable*, which is in *binary* format (as opposed to text). Released by the Free Software Foundation, gcc is a Unix-based C compiler usually operated via the command line. It often comes distributed with a Unix installation, so if you are running Unix or a Linux variant you likely have it on your system. Well, understanding this isn't necessary to programming, and if you're confused, you'll pick it up later on. You can invoke gcc on a C source code file simply by typing *gcc filename.c*

```
[CPSEC1@localhost ~]$ gcc sinterest.c
```

Your *.c* file will compile and gets executed, if there are no errors. If there are errors, then the list of these will be displayed when you type the above command and press enter. Kindly get rid of these errors to proceed. Now type ls on the \$ prompt and observe that a new file is been created.

```
[CPSEC1@localhost ~]$ ls
a.out sinterest.c
```

By default the name of the executable file will be *a.out*, to see the output, type the following and observe the output of your C program.

```
[CPSEC1@localhost ~]$ ./a.out
```

Now, let us dissect our above program line by line. Any C program is nothing but a combination of functions. One such function is **main(void)**, and the execution of your C program starts with this function. Parenthesis after main is necessary. Set of curly braces which contain the body of your main function. Any variable used in the program must be declared before using it. First two lines in the braces do the same and declare two variables of type integer and float each. After the execution of the assignment statement (**si=pri\*time\*rate/100**) variable **si** contains the simple interest.

**printf()** is a function which is used to display the output print on the screen. The general form is:

```
printf("<format string>", <list of variables>;
<format string could be>,
%d for printing integer values.
%f for printing real values.
%c for printing character values.
```

A message also can be included along with format string as shown in the above program. The message is "The Calculated Simple Interest is" followed by "%f". The message will be printed as it is and as soon as %f will be observed, the value of the variable after comma will be printed at its place.

Finally the **return (0)** statement takes you out to the prompt.

### 3. Options for controlling the kind of output

Compilation of a C program can involve up to four stages in a predefined sequence: preprocessing, compilation proper, assembly and linking. Different compilation options can be used to instruct at which stage gcc should stop. The most commonly used compilation options are:

- 1) -E: Stop after the preprocessing stage and do not run the compiler proper. The output is in the form of preprocessed source code, which is sent to the standard output.

Type the following and observe the output:

**[CPSEC1@localhost ~]\$gcc -E sinterest.c**

- 2) -S: Stop after the stage of compilation proper and do not assemble. The output is in the form of an assembler code file for each input file specified. By default, the assembler file name for a source file ends with '.s' extension.

Type the following and observe the output:

**[CPSEC1@localhost ~]\$gcc -S sinterest.c**

- 3) -c: Compile or assemble the source files, but do not link. The output is in the form of an object file for each source file. By default, the object file name for a source file ends with '.o' extension.

Type the following and observe the output:

**[CPSEC1@localhost ~]\$gcc -c sinterest.c**

- 4) -o <filename>: The default executable output of gcc is "a.out". It is also possible to specify a name for the executable file at the command line by using the syntax  
`gcc filename.c -o outputfile.out`

Type the following:

**[CPSEC1@localhost ~]\$gcc sinterest.c -o sinterest.out**

So, now the output file is *sinterest.out* which you can execute by following command

**[CPSEC1@localhost ~]\$ ./sinterest.out**

### Try out this modifications

1. Replace the initialization statements in the given program with the following:

```
printf ("Enter principal amount:");
scanf ("%d", &pri);
printf ("Enter time:");
scanf ("%d", &time);
printf ("Enter rate of interest:");
scanf ("%f", &rate);
```

**scanf()** function is used to take the input from the user and store it in a variable. So you are actually specifying to the user to enter the values of the three variables. Execute the above modified program using procedure explained above.

2. Write a C program that takes two numbers from the user as an input and it calculates and displays the sum, multiplication, division and subtraction of these numbers. Assume that user enters non zero numbers.

*Hint: Declare two integer variables to store the numbers and four integer variables for storing sum, multiplication, division and subtraction result. Read the numbers using scanf(). Write arithmetic expressions for the required result. Use printf() to display the results.*

*Note: Modify your program for double data type.*

3. Area of a circle of radius R is given by ( $\text{Area} = \text{PI} * R * R$ ). Write a C program to calculate the area of a circle. Here PI is a constant having value 3.143

#### **4.Decisions! Decisions!**

Conditional constructs allows a programmer to select an action based on some condition.

##### **4a.if, if else construct:**

The process of displaying larger of two numbers can be described in human language as:

```

If A is greater than B
    Print A is larger
else
    Print B is larger

```

The condition will always evaluate to True/False. Here the result of the condition **A is greater than B** decides which action will be taken. C has three major decision making statements: the **if** statement, the **if-else** statement, and the **switch** statement.

**Syntax of**

**If**

```

if( condition )
{
    expr_set;
}

```

**If ..... else**

```

if ( condition )
{
    expr_set1;
}
else
{
    expr_set2;
}

```

**If ..... else if ..... else**

```

if( condition 1 )
{
    expr_set1;
}
else if( condition 2 )
{
    expr_set2;
}
else if( condition 3 )
{

```

```

        expr_set3;
    }
    else
    {
        expr_set4;
    }

```

Consider the following C program (calc.c) which exhibits operation of a simple calculator capable of performing the operation depending on the operator being supplied by the user.

```

#include <stdio.h>
#include <math.h>
int main(void)
{
    int num1, num2;
    float res;
    char op;
    printf ("Enter First Number\n");
    scanf ("%d",&num1);
    printf ("Enter Second Number\n");
    scanf ("%d",&num2);
    printf ("Enter the operator (+ , - , * , / , ^)\n");
    scanf (" %c",&op);
    if (op == '+')
        res = num1 + num2;           /*Implicit Type Casting */
    else if (op == '-')
        res = num1 - num2;
    else if (op == '*')
        res = num1 * num2;
    else if (op == '/')
        res = (float)num1/(float)num2; /*Explicit Type
                                        Casting */
    else if (op == '^')
        res = pow(num1,num2);
    else
    {
        printf ("\n Invalid Operator");
        return (0);
    }
    printf ("\n num1 %c num2 = %f",op,res);
    return (0);
}

```

Observe that in the above program, res is declared as float. So in the expression “res = num1 + num2 “, first the RHS of the expression gets evaluated (which is an addition of two integers), the resulting integer value gets changed to float *implicitly* (because the LHS is float), and assignment is done. Similar explanation is true for subtraction and multiplication. As num1 and num2 declared as integers so num1/num2 becomes an integer division. (i.e. fractional part of the result will be truncated) One of the operand either num1 or num2 temporarily needs to be converted into float or double to get the correct result. This is called explicit type casting. Also, we are using the standard library function, pow(), to find num1 raised to power num2. This function is defined in math.h standard library.

Compile and run the program for various possible inputs.

#### 4b.Switch-case construct:

The **switch-case** statement is a multi-way decision statement. Unlike the multiple decision statement that can be created using if-else, the **switch** statement evaluates the conditional expression and tests it against numerous constant values. The branch corresponding to the value that the expression matches is taken during execution.

The value of the expressions in a switch-case statement must be an ordinal type i.e. integer, char, short, long, etc. Float and double are not allowed.

Syntax of Switch ..... case

```

switch( expression )
{
    case constant-expression1:
        expr_set3;
        break;
    case constant-expression2:
        expr_set3;
        break;
    case constant-expression3:
        expr_set3;
        break;
    default:
        expr_set3;
        break;
}

#include <stdio.h>
#include <math.h>
int main(void)
{
    int num1, num2;
    float res;
    char op;
    printf ("Enter First Number\n");
    scanf ("%d",&num1);
    printf ("Enter Second Number\n");
    scanf ("%d",&num2);
    printf ("Enter the operator (+ , - , * , / , ^)\n");
    scanf (" %c",&op);
    switch (op)
    {
        case '+':
            res = num1 + num2; /*Implicit Type
                                Casting */
            break;
        case '-':
            res = num1 - num2;
            break;
        case '*':
            res = num1 * num2;
            break;
        case '/':
            res = (float)num1/(float)num2;

```

```

        /*Explicit Type Casting */
        break;
    case '^':
        res = pow(num1,num2);
        break;
    default:
        printf ("\n Invalid Operator");
        return (0);
}
printf ("\n num1 %c num2 = %f",op,res);
return (0);
}

```

## **5.Exercises:**

### **1.Write a program to display menu in the restaurant**

First take time as input.

M-Morning

A-Afternoon

E-Evening

If timing is morning then display the morning menu as:

‘Good morning!!! please select item’

1. Tea
2. Coffee
3. Sandwich
4. Fruit Dish

For Selected item print the price of that item.

1. Tea -Rs.20
2. Coffee-Rs.20
3. Sandwich-Rs.30
4. Fruit Dish-Rs.30

If timing is afternoon then display the afternoon menu with

‘Good afternoon!!! please select item’

1. Rice Combo
2. Veg Combo
3. Non veg Combo

For Selected item print the price of that item.

1. Rice Combo : 50
2. Veg Combo : 60
3. Non veg Combo : 70

If timing is evening then display the evening menu with

‘Good evening!!! please select item’

1. Juice
2. Milkshake

For Selected item print the price of that item.

1. Juice : 20
2. Milkshake : 30

```
#include<stdio.h>
int main()
{
    char time;
    int choice;
    printf("Enter current time");
    printf("\n M.Morning");
    printf("\n A.Afternoon");
    printf("\n E.Evening\n");
    scanf("%c",&time);
    if(time=='M')
    {
        printf("Good morning!!! please select item");
        printf("\n 1.Tea");
        printf("\n 2.Coffee");
        printf("\n 3.Sandwitch");
        printf("\n 4.Fruit Dish");
        scanf("%d",&choice);
        switch (choice) {
            case 1:
                printf("Price of tea is 20");
                break;
            case 2:
                printf("Price of coffee is 20");
                break;
            case 3:
                printf("Price of Sandwitch is 30");
                break;
            case 4:
                printf("Price of Fruit Dish is 30");
                break;
            default:
                printf("please select correct option");
                break;
        }
    }
}
```



```

}
else if(time=='A')
{
    printf("Good afternoon!!! please select item");
    printf("\n 1.Rice Combo");
    printf("\n 2.Veg Combo");
    printf("\n 3.Nonveg Combo");
    scanf("%d",&choice);
    switch (choice) {
        case 1:
            printf("Price of Rice Combo is 50");
            break;
        case 2:
            printf("Price of Veg Combo is 60");

            break;
        case 3:
            printf("Price of NonVeg Combo is 70");
            break;
        default:
            printf("please select correct option");
            break;
    }
}
else if(time=='E')
{
    printf("Good evening!!! please select item");
    printf("\n 1.Juice");
    printf("\n 2.Milkshake");
    scanf("%d",&choice);
    switch (choice) {
        case 1:
            printf("Price of juice is 20");
            break;
        case 2:
            printf("Price of milkshake is 30");
            break;
        default:
            printf("please select correct option");
            break;
    }
}
return 0;
}

```

## 2. Write a program to calculate different factors in salary of the employee

Take input as basic salary and calculate Dearness allowance, HRA, insurance, Provident Fund , Gross Salary, Net Salary by using below formulas

Dearness allowance=basic\*0.20,

HRA=basic\*0.15,  
Insurance=500,  
Provident Fund=basic\*0.12 ,  
Gross=basic+da+hra  
Net=gross-(insurance+pf)

```
#include<stdio.h>
int main()
{
    float insurance,basic,gross,net,hra,da,pf,insu;
    int choice;
    printf("Enter basic salary");
    scanf("%f",&basic);
    printf("Enter choice for calculating salary factors");
    printf("\n 1.Dearness Allowance");
    printf("\n 2.HRA");
    printf("\n 3.Provident Fund");
    printf("\n 4.Gross Salary");
    printf("\n 5.Net Salary");
    printf("\n 6.insurance\n");
    scanf("%d",&choice);
    switch (choice) {
        case 1:
            da=basic*0.20;
            printf("Dearness Allowance is %f",da);
            break;
        case 2:
            hra=basic*0.15;
            printf("HRA is %f",hra);
            break;
        case 3:
            pf=basic*0.12;
            printf("Provident Fund is %f",pf);
            break;
        case 4:
            da=basic*0.20;
            hra=basic*0.15;
            gross=basic+da+hra;
            printf("Gross Salary is %f",gross);
            break;
        case 5:
            da=basic*0.20;
            hra=basic*0.15;
            gross=basic+da+hra;
            pf=basic*0.12;
            insu=500;
            net=gross-(insurance+pf);
            printf("Net Salary is %f",net);
            break;
```

```

        case 6:
            insu=500;
            printf("Insurance is %f",insu);
            break;
        default:
            printf("please select correct option");
            break;
    }
    return (0);
}

```

3. Write a program to check the entered character is vowel or not.

```

#include<stdio.h>
int main()
{
    char alpha;
    printf("Enter a character");
    scanf("%c",&alpha);
    switch (alpha) {
        case 'a':
            printf("Character is a Vowel");
            break;
        case 'e':
            printf("Character is a Vowel");
            break;
        case 'i':
            printf("Character is a Vowel");
            break;
        case 'o':
            printf("Character is a Vowel");
            break;
        case 'u':
            printf("Character is a Vowel");
            break;
        case 'A':
            printf("Character is a Vowel");
            break;
        case 'E':
            printf("Character is a Vowel");
            break;
        case 'I':
            printf("Character is a Vowel");
            break;
        case 'O':

```

```

        printf("Character is a Vowel");
        break;
    case 'U':
        printf("Character is a Vowel");
        break;
    default:
        printf("Character is not Vowel");
        break;
    }
    return 0;
}

```

## **Few more exercises,**

- 1) Write a C program that takes month number as an input and displays the number of days in the month. If input is greater than 12 or less than 1 then program should display an error message.
- 2) Write a c program to find the solution for the given equation.

$$X = \frac{-b \pm (b^2 + 2 \cdot 4ac)}{2a}$$

- 3) Evaluate the following expression if abc=2.5, big=2 and g is a double

$$G = \text{big} / 2 + \text{big} * 4 / \text{big} - \text{big} + \text{abc} / 3$$

- 4) The current date, month, year and the date, month, year in which the employee joined in the organization are entered through the keyboard. You are asked to calculate the service of the employee in the organization. The service should be displayed as Number of years, Number of months completed in the organization.
- 5) The marks obtained by a student in 5 different subjects are input through the keyboard. The student gets a division as per the following rules:

Percentage above or equal to 60-First division

Percentage between 50 and 59-2nd division

Percentage between 40 and 49-3rd division

Percentage less than 40 –Fail

Write a C program for the above conditions.

- 6) A library charges a fine for every book returned late. For first 5 days the fine is 5 rupees, for 6-10 days fine is 100 rupees and above 10 days fine is 500 rupees. If you return book the book after 30 days your Membership will be cancelled and your caution deposit (Rs 5000) will be taken as fine. Write a C program to accept the number of days the member is late to return the book and display the fine amount.