Programming Fundamentals CT-175

Constants, Variables and Operators

Objectives

The objective of this lab is to enable students to understand the basic data types supported by C, rules for defining variables, and writing sequential programs by making use of format specifiers. By the end of this lab students will be able to write simple sequential programs including variables, inputs, processes, and outputs.

Tools Required

DevC++ IDE

Course Coordinator –
Instructor –
Lab Instructor –
Department of Computer Science and Information Technology
NED University of Engineering and Technology

Variables in C

In programming, a variable is a container (storage area) to hold data. To indicate the storage area, each variable should be given a unique name (identifier). Variable names are just the symbolic representation of a memory location. A variable can be declared and initialized using following syntax:

```
datatype   variable_name   =   value ;
```

Example:

```
int num = 9;
```

Here, num is a variable of integer type. Here, the variable is assigned an integer value 9.

Rules for naming a variable

- ✓ A variable name can only have letters (both uppercase and lowercase letters), digits and underscore.
- ✓ The first letter of a variable should be either a letter or an underscore.
- ✓ There is no rule on how long a variable name (identifier) can be. However, you may run into problems in some compilers if the variable name is longer than 31 characters.
- ✓ C is a statically typed language. This means that the variable type cannot be changed once it is declared.

Basic Data Types Supported by C

In C programming, data types are declarations for variables. This determines the type and size of data associated with variables. Here is a table containing commonly used types in C programming for quick access.

C Basic Data Types	32-bit CPU		64-bit CPU	
	Size (bytes)	Range	Size (bytes)	Range
char	1	-128 to 127	1	-128 to 127
short	2	-32,768 to 32,767	2	-32,768 to 32,767
int	4	-2,147,483,648 to 2,147,483,647	4	-2,147,483,648 to 2,147,483,647
long	4	-2,147,483,648 to 2,147,483,647	8	-9,223,372,036,854,775,808-9,223,372,036,854,775,807
long long	8	9,223,372,036,854,775,808-9,223,372,036,854,775,807	8	9,223,372,036,854,775,808-9,223,372,036,854,775,807
float	4	3.4E +/- 38	4	3.4E +/- 38
double	8	1.7E +/- 308	8	1.7E +/- 308

Format Specifiers

Format Specifiers are strings used in the formatted input and output. The format specifiers are used in C to determine the format of input and output. Using this concept the compiler can understand what type of data is in a variable while taking input using the scanf( ) function and printing using printf( ) function.

Some of the commonly used Format Specifiers are given in the following Table.

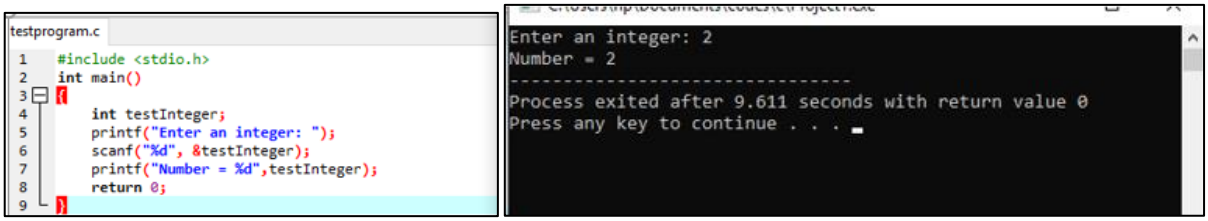
Format specifier	Description
`%d` or `%i`	It is used to print the signed integer value where signed integer means that the variable can hold both positive and negative values.
`%u`	It is used to print the unsigned integer value where the unsigned integer means that the variable can hold only positive value.
`%o`	It is used to print the octal unsigned integer where octal integer value always starts with a 0 value.
`%x`	It is used to print the hexadecimal unsigned integer where the hexadecimal integer value always starts with a 0x value. In this, alphabetical characters are printed in small letters such as a, b, c, etc.
`%X`	It is used to print the hexadecimal unsigned integer, but %X prints the alphabetical characters in uppercase such as A, B, C, etc.
`%f`	It is used for printing the decimal floating-point values. By default, it prints the 6 values after '.'.
`%e` or `%E`	It is used for scientific notation. It is also known as Mantissa or Exponent.
`%g`	It is used to print the decimal floating-point values, and it uses the fixed precision, i.e., the value after the decimal in input would be exactly the same as the value in the output.
`%p`	It is used to print the address in a hexadecimal form.
`%c`	It is used to print the unsigned character.
`%s`	It is used to print the strings.
`%ld`	It is used to print the long-signed integer value.

Input in C

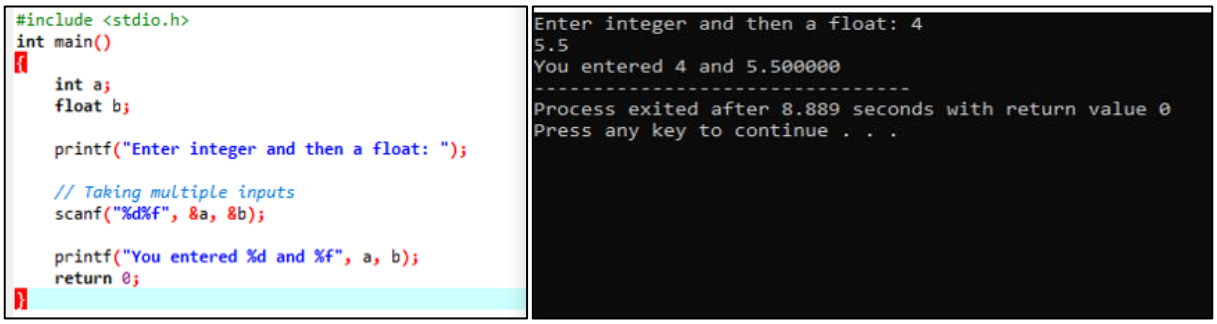
In C programming, **`scanf( )`** is one of the commonly used function to take input from the user. The **`scanf( )`** function reads formatted input from the standard input such as keyboards. The syntax of **`scanf( )`** function is:

```
scanf("format string",&variable_name);
```

Eample 01: Single input from user…



Example 02: Multiple inputs from user …



Output in C

In C programming, printf( ) is one of the main output function. The function sends formatted output to the screen. The syntax of printf( ) function is:

```
printf("format string",variable_name);
```

Example 03:

```c
#include <stdio.h>

int main(){

    printf("Hello World");

}
```

```
Hello World
--------------------------------
Process exited after 0.01564 seconds with return value 0
Press any key to continue . . .
```

Example 04:

```c
#include <stdio.h>

int main(){

    int testInteger=5;
    printf("Number is %d",testInteger);

}
```

```
Number is 5
--------------------------------
Process exited after 0.03464 seconds with return value 0
Press any key to continue . . .
```

Exercise

1. Write a C program that takes two integer values as input from the user. Then swap the values taken from the user and display the output of the variables.
2. A customer asks the IT firm to develop a program in C language, which can take tax rate and salary from the user on runtime and then calculate the tax, the user has to pay and the salary he/she will have after paying the tax. This information is then provided to the user.
3. A car traveled for some hours. The time car traveled is taken at run time of the program, and it must not be negative and must be between one to five hours. The car had not traveled same distance in each hour. The distance that the car covered must not be negative. Write a C Program that computes the Average Speed of the Car in miles per hour. Hint: the restrictions can be displayed in the form of message on the window.
4. Explain the output of this C program. Why the wrong value is being displayed in the output?

```c
#include <stdio.h>

int main(){

    int testInteger=3000000000;
    printf("Number is %d",testInteger);

}
```

```
D:\PF\Lab 3 example codes\Printf.exe
Number is -1294967296
--------------------------------
Process exited after 0.03059 seconds with return value 0
Press any key to continue . . .
```

5. Construct a C program with the flowchart below. The input value of the Principle must be between 100 Rs. To 1,000,000 Rs. The Rate of interest must be between 5% to 10% and Time Period must be between 1 to 10 years. Hint: these restrictions can be displayed in the form of message on the window.