Laboratory Activity 5

Introduction to C- Programming

Objectives:

1. Learn the basic of C-Language: printf() and scanf().
2. Develop algorithms and flowcharts for use in programming applications.
3. Design, compile, test, run, and implement C language program

Discussion:

In the Beginning

- ✓ C is a general purpose programming language and it started to run under UNIX environment.
- ✓ Dennis Ritchie invented the C-Programming language at AT&T Bell Laboratories
- ✓ C is similar to Pascal, a functional programming language
- ✓ Aimed in designing and developing operating systems, compilers, interpreters, database software and embedded intelligence programs.

C Data Types

Before a variable name can be used in a C program, it must be declared explicitly, together with its corresponding data type.

Variable Type	Keyword	Bytes Required	Range
Character	char	1	-128 to 127
Unsigned character	unsigned char	1	0 to 255
Integer	int	2	-32768 to 32767
Short Integer	short int	2	-32768 to 32767
Long Integer	long int	4	-2,147,483,648 to 2,147,438,647
Unsigned Integer	unsigned int	2	0 to 65535
Unsigned Short integer	unsigned short int	2	0 to 65535
Unsigned Long Integer	unsigned long int	4	0 to 4,294,967,295
Float	float	4	1.2E-38 to
Double	double	8	2.2E-308 to
Long Double	long double	10	3.4E-4932 to 1.1E+4932

Variable Declaration

Before you can use a variable, you must tell the compiler about it by declaring it and telling the compiler about what its "type" . To declare a variable you use the syntax:

<variable type> <name of variable>;

- ✓ Individual declaration

```
int sum;

float area;

char letter;
```

- ✓ Group declaration

```
int n1, n2, sum, diff;

float area, fahr, celsius, div;
```

A Simple Program

```
#include <stdio.h>

int main()

{

   printf( "ANIMO La Salle!\n" );

   getchar();

   return 0;

}
```

The #include is a "preprocessor" directive that tells the compiler to put code from the header called stdio.h into the program before actually creating the executable. By including header files, you can gain access to many different functions--both the printf and getchar functions are included in stdio.h.

The next important line is int main(). This line tells the compiler that there is a function named main, and that the function returns an integer, hence int. The "curly braces" ({ and }) signal the beginning and end of functions and other code blocks.

The printf function is the standard C way of displaying output on the screen. The quotes tell the compiler that you want to output the literal string as-is (almost). The '\n' sequence is actually treated as a single character that stands for a newline; for the time being, just remember that there are a few sequences that, when they appear in a string literal, are actually not displayed literally by printf and that '\n' is one of them. The actual effect of '\n' is to move the cursor on your screen to the next line. Notice the semicolon: it tells the compiler that you're at the end of a command, such as a function call. You will see that the semicolon is used to end many lines in C.

The next command is getchar(). This is another function call: it reads in a single character and waits for the user to hit enter before reading the character. This line is included because many compiler environments will open a new console window, run the program, and then close the window before you can see the output. This command keeps that window from closing because the program is not done yet because it waits for you to hit enter. Including that line gives you time to see the program run.

Finally, at the end of the program, we return a value from main to the operating system by using the return statement. This return value is important as it can be used to tell the operating system whether our program succeeded or not. A return value of 0 means success. The final brace closes off the function. (Source: http://www.cprogramming.com)

printf() and scan()

printf()

This function is used in order to display the contents of a variable on the screen or to simply print the contents as it is on the screen. This function can be accessed from anywhere within a program simply by writing the function name, followed by a list of arguments enclosed in parentheses.

The arguments represent data items that are sent to the function. The necessary information required for printf ( ) and scanf ( ) function is supplied by < stdio.h > file. This < stdio.h> file is called header file. Different header files are like < conio.h >,< stdio.h > etc.

The general form of printf ( ) is

printf("format string"); or

printf(" < format string >" , < list of variables>);

< Format string > consists of character to be display with following specifiers.

%f: For printing real values. %d:
For printing integer values.
%c: For printing character values.
%s: For printing string values.

scanf ( )

Input is entered into the computer from a standard input device by using scanf ( ) function. This function can be used to enter any combination of numerical values, single characters & strings.

The general form of scanf ( ) is

scanf("data type format", &variable name);

Note: that the ampersand (&) before the variables in the scanf ( ) statement is must

Basic Data Type Format

Type	Format	Meaning
int	%d	whole number (3, 7, -9, 10000)
float	%f	A number with decimal point (5.8, -5.67,0.32)
char	%c	A single letter, symbol or number enclosed with single quotes ('B', 'r', '*')
	%s	a string of characters enclosed with double quotes ("Bok", "JAY")

Example 1: Write a C program that calculates the sum of two input numbers and display the result

Algorithm:
Input: Enter two numbers (n1, n2)
Process: Compute the sum (sum=n1 +n2)
Output: Display the sum (sum)

```
#include   <stdio.h>
#include  <stdlib.h>
int main()
{
  int sum, n1, n2;
   printf("\n Enter two numbers: ");
  scanf("%d %d", &n1, &n2);
  sum = n1+n2;
  printf("\n The sum is: %d\n", sum);
  system("PAUSE");
  return 0;
}
```

Example 2: Write a program to calculate the area of a circle and display the result. Use the formula:
$A = \pi r^2$

Algorithm:
Input: enter the radius
Process: Compute the Area
Output: Display the area

```
// 2. Area of a Circle
#include <stdio.h>
#include <stdlib.h>
#define Pi 3.1416
int main()
        {
        int r;
        float A;
                printf("\nEnter the radius:" );
                scanf("%d", &r);

                A=Pi*r*r;
                printf("The area: %5.2f \n", A);

                system("PAUSE");
                return 0;
        }
```

Example 3: Write a program that computes the average of three input quizzes, then display the result.

Algorithm
Input: Enter three quizzes(q1, q2, q3)
Process: Compute the average (ave=(q1+q2+q3)/3)
Output: Display the Average

```
#include   <stdio.h>
#include  <stdlib.h>
int main()
{
int q1, q2,q3;
float ave;
        printf("\nEnter Three Quizzes: ");
        scanf("%d %d %d", &q1, &q2, q3);
        ave=(q1+q2+q3)/3;
        printf("the average: %.3f\n", ave);
 system("PAUSE");
 return 0;
}
```