Certificate - Software Development

Input Output Functions

bcs Accredited Course Provider
© All rights reserved

Input and output Functions

- In C language to perform input and output operations there are built in function that comes with the C language, which are called **standard library functions**. Here we will be learning the printf() and the scanf() functions.
  - **printf() function** - This is used to print a formatted on the screen including character, string, float, integer, octal and hexadecimal values
  - **scanf() function** - This is used to read formatted input from the keyboard containing character, string and numeric data

© All rights reserved

printf() Function

Syntax:

printf(“Control String”, arg1, arg2, …. argN);

According to the syntax everything that is given in the control string is printed exactly as it is except for % signs (percentage directive) and \ signs (backslash directive) appearing within that.

- (**%**) Percentage directive – is used to print data values
- (****) Backslash directive – is used to control characters

printf() is defined in the header file **stdio.h** therefore must include the following statement when calling it from a program

 # include <stdio.h>

bcs Accredited Course Provider
© All rights reserved

% Directive in printf()

% Directive is used to print data values	
% Directive	**Description**
%d	argument is printed as a decimal
%x	argument is printed as a hexadecimal
%o	argument is printed as an octal value
%f	argument is printed as a fractional number
%c	argument is printed as a character
%s	argument is printed as a string
%%	This is used to print a percentage

© All rights reserved

\ Directive in printf()

\ Directive is used to print control characters	
\ Directive	**Description**
\n	prints a “newline” character
\b	outputs a backspace character
\r	outputs a carriage return (without a line feed)
\’	prints a single quote
\”	prints a double quote
\\	prints a single backslash

© All rights reserved

Formatting with printf()

- Any integer data type can be printed using %c, %d, %o and %x
  - %**N**d – N indicates the size of the field (leading spaces filled with spaces)
  - %**0N**d - Same as above leading spaces filled with Zeros
  - %**-N**d – Number will be left aligned spaces are kept after the number
- %**N**s will print a string using a field size of N, string will be right aligned with leading spaces.
- %**-N**s print a string using a field size of N, string will be left aligned with trailing spaces.
- %**N.M**f –N here is the total field size M is the number of decimal places

bcs Accredited Course Provider
© All rights reserved

printf() Example

```
main()
{
char ch = 'A';
short n1 = 59;
char name[] = "Esoft";
long n2 = 0xAF0;
float pi = 3.14;
printf("Value of %c is %d\n", ch, ch);
printf("\tCampus is %s Colombo\n", name);
printf("This line will be overridden\r");
printf("Campus is %-10s Colombo\n", name);
printf("%d(Dec)= %o(Oct)= %x(Hex)\n", n1, n1, n1);
printf("%-6x(Hex)= %06d(Dec)\n", n2, n2);
printf("Value of pi = %f\n", pi);
printf("Value of pi = %05.1f\n", pi);
}
```

Program Output
Value of A is 65
Campus is Esoft Colombo
This line will be overridden
Campus is Esoft Colombo
59(Dec)= 73(Oct)= 3b(Hex)
Af0 (Hex)= 002800(Dec
Value of pi = 3.140000
Value of pi = 003.1

© All rights reserved

scanf() Function

- Syntax: **scanf("Control String", arg1, arg2, …. argN);**
- This function is used to input data from the keyboard. Correct % directive must be used depending on the type of data.
- If the input data doesn't match the data format given, it will incorrectly read values.

Reading a Character from the Keyboard
`char ch;`
`printf("Enter a Character : ");`
`scanf("%c", &ch);`

Accredited Course Provider
© All rights reserved

scanf() – More examples

- If the input data doesn't match the data format given, it will incorrectly read values.
- **fflush(stdin);** - safer to use this to clear the input buffer, to prevent remaining buffer content incorrectly being read as the next input.

Reading a Float number from the Keyboard

```
float rate;
printf("Enter inerest rate : ");
scanf("%f", &rate);
```

Reading a series of Integers

```
int d, m, y;
printf("Enter your DOB (dd-mm-yy) : ");
scanf("%d-%d-%d", &d, &m, &y);
```

Reading a String from the Keyboard

```
char name[30];
printf("Enter your Name : ");
scanf("%s", name);
```


Accredited Course Provider
© All rights reserved

Sample Program

- Following example demonstrates scanf(), printf() and the some of the operators. It calculates and displays the area and the perimeter of a circle when the radius is entered from the keyboard.

```
main()
{
        int radius;
        float area, peri;
        const float pi = 3.14;
        printf("Enter radius : ");
        scanf("%d",&radius);
        area = pi * radius * radius;
        peri = 2 * pi * radius;
        printf("Area = %.2f\n",area);
        printf("Perimeter = %.2f\n",peri);
}
```

Program Output
Enter radius : 5
Area = 78.50
Perimeter = 31.40

bcs Accredited Course Provider
© All rights reserved

Lesson Summary

- Standard Library Functions
- printf() Function
- Percentage Directive
- Backslash Directive
- Formatting with printf()
- Sample program for printf()
- scanf() Function

© All rights reserved