Managing Input Output Operators

Input and Output operations are used to perform the function of supplying the input data to a program and obtaining the output results.

Input means to provide the program with some data to be used in the program and Output means to display data on screen or write the data to a printer or a file.

C programming language provides many built-in functions to read any given input and to display data on screen when there is a need to output the result.

All these built-in functions are present in C header file stdio.h

Standard Input and Output functions

1. **Formatted** **2. Unformatted**

Formatted:

1. **scanf(): is used to accept input data from the standard device ie. Keyboard in a fixed format.**

   Syntax : scanf("format string", address of operator variable);

   Format string: contains format specifier which starts from % character followed by datatype alphabet.

Format specifier	Input types	Data types
%d %u %ld %lu	signed int unsigned int long signed int long unsigned int	Integers
%f %lf %Lf	float double long double	Float
%c	Signed / unsigned character	%c
%s	String	String

Arguments depend on the type of the data type and is preceded by the &

Eg: scanf("%d %f %c", &a,&b,&c);

2. **printf(): is used to output data from the computer onto a standard output device in a fixed format.**

   Syntax: printf(" format string", arguments)

Format string contains:

1. **Characters to be printed.**
2. **Format specifier that begin with % sign.**
3. **Escape sequence**

Escape sequence	use
\n	New line
\t	Tab space
\r	Carriage return
\f	formfeed

Options of format specifier

D %5d

. %7.2f 1234.45

- Left justification of the output

Unformatted console I/O functions:

getchar() getche() getch() putchar()	1. Read a single character from the standard input device 2. The read character is displayed on the screen 3. And user press the enter key 1. Reads a single character 2. Character is displayed 3. User does not press enter key 1. Reads a character 2. The character is not echoed 3. The user is not required to press the enter key 1. Used to send a single character to the standard output device	Character I/O
gets()	1. Reads a string from input standard device 2. The length of the string is limited to string variable 3. The string ends with enter key 1. Used to send s string to the standard output device.	String I/O

puts()		

Eg:

```
// program to demonstrate character I/O function
#include<stdio.h>
#include<conio.h>
void main()
{
 char ch;
 clrscr();
 printf("\n press any key to continue");
 getch();
 printf("\n enter any character");
//or
puts("\n enter any character");
 ch=getche();
 printf("\n entered character is ");
 putchar(ch);
 printf("\n enter another character");
 ch=getchar();
 printf("\n entered character is ");
 putchar(ch);
 getch();
}
// program to demonstrate string I/O function
#include<stdio.h>
#include<conio.h>
void main()
{
 char n[20];        //character array variable size is 20
 clrscr();
 printf("\n enter your name");
 gets(n);                    or        n=gets();
 printf("\n your name is");
```

```
puts(n);
getch();}
```

END