

Input & output Statement

It is called as "Library functions".

Input Functions:

This fun. are used to accept input data from the users. Eg: Getting data from keyboard.

O/p. Function:

This fun. are used to display the O/P. to the user.

eg: Printing O/P using monitor.

The I/p & O/p Fun. are classified into two types.

1. Formatted function.

2. Unformatted function.

getch()
→ getche()
→ getchae()
→ gets()
→ putch()
→ putchar()

Formatted Function:

① **printf**: displays the O/P on screen.
→ It is used for formatted O/P.

general form is

printf (control string, argument list);

Control string $\rightarrow$ a string that contains formatting information & ~~the~~ argument

Argument list $\rightarrow$ the data items that will be printed

The printf function also contains conversion specification that begin with a % sign.

It also contains conversion specification that begin with \ sign.

eg:

```
main()
```

```
{ int avg = 450;
```

```
float per = 75.5;
```

```
printf ("Average = %d In percentage = %.f", avg, per)
```

```
}
```

O/p:

Average = 450

percentage = 75.50000.

%d $\rightarrow$ denotes average that will be printed integer.

%.f $\rightarrow$ denotes percentage " floating point val

\n $\rightarrow$ used. beginning of the next line.

scanf: take input from user.

$\rightarrow$ I/p formatted.

$\rightarrow$ This fun. accept the input from keyboard using control variables.

general form is

scanf (Control string, argument list);

→ The control string also contains conversion specification.

→ It specifies the format in which numeric data is to be accepted. If more than one control string is used, then the same number of arguments should also be used in the functions.

Ex:

```
void main()
{
    int a, b;
```

```
    float c;
```

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

```
}
```

a, b, c → variables

& → address of operator.

a, b → integer variable, specifies %d.

c → float variable, %f for conversion.

```
(ii) int a;
```

```
scanf("%d", &a);
```

This statement is used to read.

Integer	short Signed short unsigned. long Signed long unsigned. unsigned hexadecimal. unsigned octal.	%d (or) %i %u %ld %lu %x %o
Real	float double.	%f %lf
Character (Single char)	Signed character unsigned character	%c %C
String		%s

Unformatted I/O functions

I/O fun:

1. getch()

2. getche()

3. getchar()

4. gets()

getch() → The returns the char. that the user has typed without displaying it on the screen.

getche() → displays it on the screen.

getchar() → This fun. is used to read a single char. from the keyboard. It requires enter key to be

gets() → types following the char.

↳ This fun. is used to read a string of char. until a new line char "\n" is entered.

data type		conversion char.
Integer	short signed short unsigned long signed long unsigned unsigned hexadecimal unsigned octal	%d (or) %i %u %ld %lu %x %o
Real	float double	%f %lf
Character (single char)	Signed character unsigned character	%c
String		%s

Unformatted I/O functions

I/O fun:

1. getch()
2. getche()
3. getchc()
4. getcs()

getch() → The returns the char. that the user has typed without displaying it on the screen.

getche() → displays it on the screen.

getchc() → This fun. is used to read a single char. from the key board. It requires enter key to be

getcs() → types following the char.

↳ This fun. is used to read a string of char. until a new line char "\n" is entered.

Ex: main()

```
{  
    char ch;  
    printf("In press any key to continue");  
    getch();  
    printf("In type any character");  
    ch = getch();  
    printf("In type any character");  
    getch();  
    printf("In continue Y/N");  
}
```

O/p

Press any key to continue

Type any character A

Type any character B

Continue Y/N N.

unformatted o/p function:

putch():

It is used to display a single char on the screen.

Ex:

char n;

n = getch();

putchar(n);

putchar():

This fun. is same as 'putch()'.
It is used to display a single char on the screen.

puts():

This fun. is used to display string on the screen.

Ex:

```
main()
```

```
{
```

```
char name [10];
```

```
puts("Enter name");
```

```
gets(name);
```

```
puts(name);
```

```
}
```

o/p:

Enter name

Raja.