

PROGRAMMING FOR PROBLEM SOLVING -- PPS SUBJECT CODE- BTPS-101-18



*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

UNIT- 2

OPERATORS IN "C"

By
Dr.A.Deepa,AP,Chandigarh
Engineering College

Session Objectives

- **Explain 8 Types of Operators**
- **Explain Arithmetic Operators**
- **Understand Arithmetic Expressions**
- **Explain Relational and Logical Operators**
- **Explain the Conditional Operators**
- **Explain the Comma Operators**
- **Explain Input/Output Statement**

*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

Operators Precedence and Associativity

Category	Operator	Associativity
Postfix	() [] -> . ++ --	Left to right
Unary	+ - ! ~ ++ -- (type)* & sizeof	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise XOR	^	Left to right
Bitwise OR		Left to right
Logical AND	&&	Left to right
Logical OR		Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %= >>= <<= &= ^= =	Right to left
Comma	,	Left to right

Arithmetic Operator
Relational Operator
Logical Operator
Assignment Operator
Increment/Decrement Operator
Conditional Operator
Bitwise Operator
Special Operator

By
Dr.A.Deepa,AP,Chandigarh
Engineering College

Arithmetic Operators

- ◆ Arithmetic Operators are used to perform numerical operations

Operator	Meaning	Example
+	Addition	x=10,y=5 x+y ->15
-	Subtraction	X-y -> 5
*	Multiplication	X*y -> 50
/	Division	x/y -> 2
%	Modulo Division	X%y -> 0

```
#include <stdio.h>
main ()
{
    int x,y;
    x = 5;
    y = 2;
    printf ( "The integers are          : %d & %d\n",x,y) ;
    printf ( "The addition gives        : %d\n", x+y) ;
    printf ( "The subtraction gives     : %d\n", %d\n", x-y) ;
    printf ( "The multiplication gives    : %d\n", x*y) ;
    printf ( "The division gives           : %d\n", x/ y) ;
    printf ( "The modulus gives            : %d\n", x%y) ;

    getch () ;
}
```

By Dr.A.Deepa,AP,Chandigarh Engineering College

Relational Operators

- ◆ Relational operators are used to test the relationship between two variables or constant

Operator	Meaning	Example	Result	value
<	Less Than	x=10,y=5 X<y	False	0
>	Greater Than	X>y	True	1
<=	Less than or Equal to	X<=y	False	0
>=	Greater than or equal to	X>=y	True	1
!=	Not Equal to	X!=y	True	1
==	Equal To	X==y	False	0

Logical Operators

- ◆ Logical operators are used to combine two or more relational expressions. This operator is used to test more than one condition at a time.

Operator	Meaning	Example	Result
&&	Logical And	When x=9, y=5 (y>=5) && (x=='9')	True
	Logical Or	(X>=6) (y=='a')	True
!	Logical Not	!(x>8)	False

Increment / Decrement Operators (++ , --)

Operator	Meaning	Syntax	Example	Result
++	Unary Plus	Variablename++; (or) ++ variable name;	X=10; X++;	X=11
--	Unary Minus	Variablename--; (or) -- variable name;	X=10; X--;	X=9

The Assignment Operator

- ◆ In C, the assignment operator(=) can be used for assigning a value to a variable

Syntax :

variable_name = expression;



LeftHand side variable

Assignment
operator

RightHand Side value

By
Dr.A.Deepa,AP,Chandigarh
Engineering College

Shorthand Assignment Operator

Syntax

Variablename <arithmetic Operator>=Expression;

Simple Assignment Operators	Equivalent Shorthand Assignment Operators
$x=x+1$	$X += 1$
$y=y-1$	$Y -= 1$
$z=z*(x+y)$	$Z *= (x+y)$
$Y=y/(x+y)$	$Y /= (x+y)$
$X=x\%z$	$X \% = z$

Conditional Operators (?:)

Simple conditional operations can be carried out with the conditional operator(?:)

Syntax

Expression1 ? Expression 2:expression 3

↑
Condition

↑
True Part

↑
False Part

```
#include<stdio.h>
void main()
{
int a=10,b=5,c;
C=(a>b)?a:b;
printf("The Result is %d",c);
}
```

OUTPUT
The Result is
10

BitWise Operators

Used in applications which require manipulation of individual bits within a word of memory

Operators	Meaning
~	One's Complement
<<	Left Shift
>>	Right Shift
&	Bitwise AND
!	Bitwise OR
^	Bitwise X-OR

Special Operators

Operators	Meaning	Example
,	Comma Operator	Z=(x=5,y=6,x+y)
*	Pointer indirection Operator	
&	Address Operator	scanf(“%d”,&no);
->	Arrow Operator in Structure	
.	Dot Operator in Structure	
#	String Sizing Operator (preprocessor)	#include<stdio.h>
##	Token passing Director	

*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

```
/* Bitwise Operator Examples */
```

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
int a,b,ans,and;
```

```
clrscr();
```

```
printf("\n Enter A Number");
```

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

```
b=1;
```

```
ans=a&b;
```

```
printf("\n The Result of AND Operation with 1");
```

```
if(ans==0)
```

```
printf("\n Rightmost bit is OFF");
```

```
else
```

```
printf("\n Rightmost bit is ON");
```

```
and=a/b;
```

```
printf("\n The Result of OR Operation with 1");
```

```
printf("\n Rightmost bit is ON and the result is %d",and);
```

```
getch();
```

```
}
```

*By Dr.A.Deepa,AP,Chandigarh
Engineering College*

Unformatted I/O Statement

✦ Characters can be read and written in C using the following functions.

Function	Operation
<code>getchar()</code>	Reads a character from the keyboard; waits for carriage return.
<code>getche ()</code>	Reads a character with echo; does not wait for carriage return not defined by ANSI, but a common extension.
<code>getch()</code>	Reads a character without echo; does not wait for carriage return; not defined by ANSI, but a common extension.
<code>putchar()</code>	Writes a character to the screen.
<code>get ()</code>	Reads a string from the keyboard.
<code>put ()</code>	Writes a string to the screen.

*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

String Based I/O Operations

gets() & Puts() are used to perform Input output operations on a string

syntax :

gets(variablename);

puts(variablename);

```
#include <stdio.h>

main()
{
    char str [80];
    /*takes input of max 80 chars*/

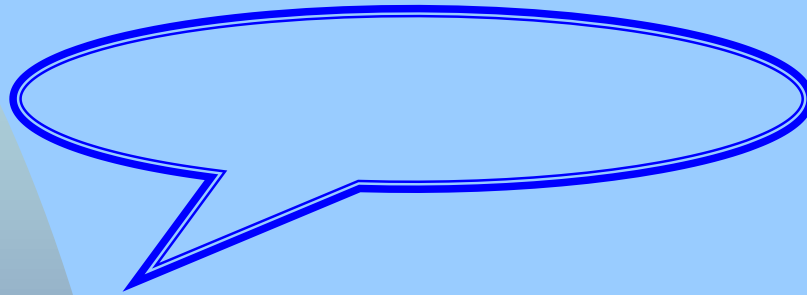
    gets (str);
    puts (str);
}
```

*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

Console Input/Output Operations

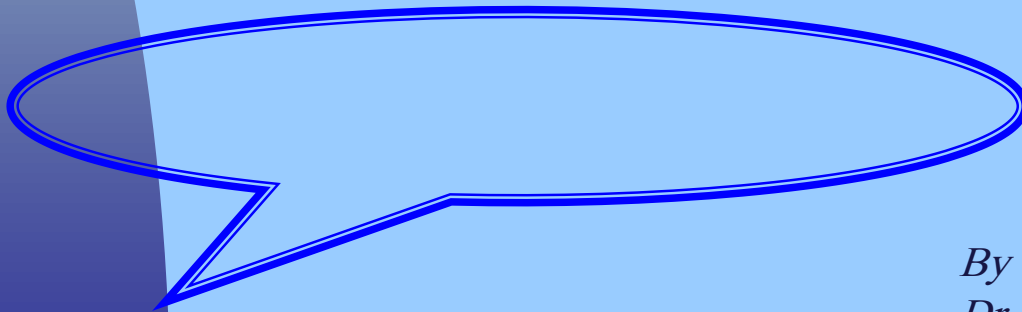
✦ All console I/O functions produce only text based outputs.

Input



```
scanf("control string",&variable1,&variable2.....);
```

Output



```
printf("control string",variable1,variable2.....);
```

*By
Dr.A.Deepa,AP,Chandigarh
Engineering College*

Format Specifier (control String)

Code	Format
<code>%c</code>	Character
<code>%d</code>	Signed decimal integers
<code>%i</code>	Signed decimal integers
<code>%e</code>	Scientific notation (lower case e)
<code>%E</code>	Scientific notation (uppercase E)
<code>%f</code>	Decimal floating point
<code>%g</code>	Uses % e or %f which ever is shorter
<code>%G</code>	Uses % E or %F, whichever is shorter
<code>%o</code>	Unsigned octal
<code>%s</code>	String of characters
<code>%u</code>	Unsigned decimal integers
<code>%x</code>	Unsigned hexadecimal (lowercase letters)
<code>%X</code>	Unsigned hexadecimal (uppercase letters)
<code>%p</code>	Displays a pointer
<code>%n</code>	The associated argument is an integer pointer into which the number of characters written so far is placed.

Scanf() and Printf() Example

```
#include<stdio.h>
#include<conio.h>
void main()
{
char empname[20];
int empno;
float salary;

printf("Enter the Employee Details")
scanf("%s %d %f",&empname,&empno,&salary)
printf("\n The employee Name is %s",empname);
printf("\n The employee Number is %d",empno);
printf("\n The employee Salary is %f",salary);
}
```

Reading decimal ,octal and hexadecimal numbers

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
clrscr();
printf("\n Enter No in decimal");
scanf("%d",&a);
printf("\n u Entered %d\n",a);

printf("\n Enter No in octal");
scanf("%o",&a);
printf("\n u Entered %o or %d in decimal\n",a,a);

printf("\n Enter No in Hexadecimal");
scanf("%x",&a);
printf("\n u Entered %x or %d\n",a,a);
getch();
}
```

*By Dr.A.Deepa,AP,Chandigarh
Engineering College*

Type Conversion

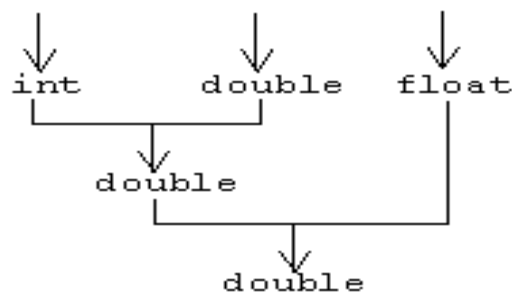
- ◆ This is used to convert one data type to another data type. The automatic type conversions for evaluating an expression are given below -

The automatic type conversions for evaluating an expression are tabulated below.

- char** and **short** are converted to **int** and **float** is converted to **double**.
- If either operand is **double**, the other is converted to **double**, and the result is **double**.
- If either operand is **long**, the other is converted to **long** the result is **double**.
- If either operand is **unsigned**, the other is also converted to **unsigned** and the result is also **unsigned**.
- Otherwise all that are left are the operands of type **int**, and the result is **int**.

- ◆ For example,

```
char ch;  
int i;  
float f;  
double d;  
result = (ch/i) + (f*d) - (f+i);
```



By
Dr.A.Deepa,AP,Chandigarh
Engineering College

The sizeof operator

- **sizeof is a unary compile-time operator**
- **The use of the sizeof operator shall be clear from the following example -**

```
#include <stdio.h>
main ()
{
    float f;
    clrscr (); /*function used to clear screen */
    printf ("Size of f is %d\n", sizeof f) ;
    printf ("Size of double is %d\n", sizeof (double)) ;
    getch () ;
}
```

Summary

- ✎ The `getchar()`, `getch()`, `getche()` deals with single character input
- ✎ The functions `gets()` and `puts()` deals with string input and output respectively
- ✎ `printf()` display any number of characters, integers, strings, float can be received at a time
- ✎ `scanf()` receives any number of characters, integers, strings, float at a time.
- ✎ `getchar()` doesnot require any argument
- ✎ `gets()` require a single argument
- ✎ In a `scanf()` strings with spaces cannot be accessed until ENTER key is pressed.
- ✎ In a `gets()` strings with any number of spaces can be accessed.

EXERCISES

1. Describe the different specifiers in scanf() function?
2. State the use of ampersand statement(s) in a scanf() statement?
3. Write a program for swapping two numbers using two variables?
4. Write a program to calculate Simple and Compound Interest?
5. Write a program to convert a decimal Number into its equivalent octal & Hexadecimal number using Format specifiers?
6. Write a program to convert temperature in centigrade to fahrenheit?
7. Write a program to find the area of the circle ($\text{area}=3.14*r^2$)?



Thank You

Queries????

By Dr.A.Deepa,AP,Chandigarh Engineering College