Example : Write a program to calculate the salary as per the following table:

Gender	Years of Service	Qualifications	Salary
Male	>= 10	Post-Graduate	15000
	>= 10	Graduate	10000
	< 10	Post-Graduate	10000
	< 10	Graduate	7000
Female	>= 10	Post-Graduate	12000
	>= 10	Graduate	9000
	< 10	Post-Graduate	10000
	< 10	Graduate	6000

```
main( )
{
      char   g ;
      int   yos, qual, sal ;

      printf ( "Enter Gender, Years of Service and
                  Qualifications ( 0 = G, 1 = PG ):" ) ;
      scanf ( "%c%d%d", &g, &yos, &qual ) ;

      if ( g == 'm' && yos >= 10 && qual == 1 )
            sal = 15000 ;
      else if ( ( g == 'm' && yos >= 10 && qual == 0 ) ||
            ( g == 'm' && yos < 10 && qual == 1 ) )
            sal = 10000 ;
      else if ( g == 'm' && yos < 10 && qual == 0 )
            sal = 7000 ;
      else if ( g == 'f' && yos >= 10 && qual == 1 )
            sal = 12000 ;
      else if ( g == 'f' && yos >= 10 && qual == 0 )
            sal = 9000 ;
      else if ( g == 'f' && yos < 10 && qual == 1 )
            sal = 10000 ;
      else if ( g == 'f' && yos < 10 && qual == 0 )
            sal = 6000 ;

      printf ( "\nSalary of Employee = %d", sal ) ;
}
```

The ! Operator

- This operator reverses the result of the expression it operates on.
- For example, if the expression evaluates to a non-zero value, then applying **!** operator to it results into a 0.
- If the expression evaluates to zero then on applying **!** operator to it makes it 1, a non-zero value .
- ! (y < 10)
- This means “not **y** less than 10”. In other words, if **y** is less than 10, the expression will be false, since **(y < 10)** is true. We can express the same condition as **(y >= 10)**.

The ! Operator

- The NOT operator is often used to reverse the logical value of a single variable, as in the expression
- if (! flag)
- This is another way of saying
- if (flag == 0)

```c
// C program to demonstrate example of
// Logical NOT (!) operator

#include <stdio.h>

int main()
{
    int num =10;

    //printing result with OR (||) operator
    printf("%d\n", !(num==10));
    printf("%d\n", !(num!=10));
    printf("%d\n", !(num>5));
    printf("%d\n", !(num<5));

    return 0;
}
```

Output

```
0
1
0
1
```

Input a year and check it is leap year or not
(it will use Logical AND (&&), Logical OR (||) and Logical NOT (!) operators).

```c
// C program to demonstrate example of
// Logical NOT (!) operator

// Input a year and check it is leap year or not
#include <stdio.h>

int main()
{
    int y;

    //input year
    printf("Enter year: ");
    scanf("%d", &y);

    //check condition
    if((y%400==0) || (y%4==0 && y%100!=0))
        printf("%d is a leap year\n",y);
    else
        printf("%d is not a leap year\n",y);

    return 0;
}
```

Output

```
First run:
Enter year: 2004
2004 is a leap year

Second run:
Enter year: 2100
2100 is not a leap year

Third run:
Enter year: 2400
2400 is a leap year

Fourth run:
Enter year: 2003
2003 is not a leap year
```

Hierarchy of Operators Revisited

Operators	Type
!	Logical NOT
* / %	Arithmetic and modulus
+ -	Arithmetic
< > <= >=	Relational
== !=	Relational
&&	Logical AND
||	Logical OR
=	Assignment

What would be the output of following program

```
main( )
{
      int   i ;

      printf ( "Enter value of i " ) ;
      scanf ( "%d", &i ) ;
      if ( i = 5 )
            printf ( "You entered 5" ) ;
      else
            printf ( "You entered something other than 5" ) ;
}
```

common mistake while using the **if** statement is to write a semicolon (;) after the condition

```
main( )
{
      int   i ;

      printf ( "Enter value of i " ) ;
      scanf ( "%d", &i ) ;
      if ( i == 5 ) ;
            printf ( "You entered 5" ) ;
}
```

working of all the three logical operators.

Operands		Results			
x	y	!x	!y	x && y	x || y
0	0	1	1	0	0
0	non-zero	1	0	0	0
non-zero	0	0	1	0	1
non-zero	non-zero	0	0	1	1

Exercise #1

Any character is entered through the keyboard, write a program to determine whether the character entered is a capital letter, a small case letter, a digit or a special symbol.

The following table shows the range of ASCII values for various characters.

Characters	**ASCII Values**
A – Z	65 – 90
a – z	97 – 122
0 – 9	48 – 57
special symbols	0 - 47, 58 - 64, 91 - 96, 123 - 127

ASCII CHART

Value	Char	Value	Char	Value	Char	Value	Char	Value	Char	Value	Char
0		22	▬	44	,	66	B	88	X	110	n
1	☺	23	↕	45	-	67	C	89	Y	111	o
2	☻	24	↑	46	.	68	D	90	Z	112	p
3	♥	25	↓	47	/	69	E	91	[	113	q
4	♦	26	→	48	0	70	F	92	\	114	r
5	♣	27	←	49	1	71	G	93	]	115	s
6	♠	28	⌐	50	2	72	H	94	^	116	t
7	•	29	↔	51	3	73	I	95	_	117	u
8	◘	30	▲	52	4	74	J	96	`	118	v
9	○	31	▼	53	5	75	K	97	a	119	w
10	◙	32		54	6	76	L	98	b	120	x
11	♂	33	!	55	7	77	M	99	c	121	y
12	♀	34	"	56	8	78	N	100	d	122	z
13	♪	35	#	57	9	79	O	101	e	123	{
14	♫	36	$	58	:	80	P	102	f	124	|
15	☼	37	%	59	;	81	Q	103	g	125	}
16	►	38	&	60	<	82	R	104	h	126	~
17	◄	39	'	61	=	83	S	105	i	127	ᴹH
18	↕	40	(	62	>	84	T	106	j	128	Ç
19	!!	41	)	63	?	85	U	107	k	129	ü
20	¶	42	*	64	@	86	V	108	l	130	é
21	§	43	+	65	A	87	W	109	m	131	â

Home Exercise #1

- If the ages of Ram, Shyam and Ajay are input through the keyboard, write a program to determine the youngest of the three.

If the ages of Ram, Shyam and Ajay are input through the keyboard, write a program to determine the youngest of the three.

Method 1

- If Ram is younger than Shyam and Ajay then Ram is the youngest
- If Shyam is younger than Ram and Ajay the Shyam is the youngest
- If Ajay is younger than Ram and Shyam then Ajay is the youngest

- If Ram is younger than Shyam
  - If Ram is younger than Ajay
    - Ram is the youngest
  - Else
    - Ajay is the youngest
- If Shyam is younger than Ram
  - If Shyam is younger than Ajay
    - Shyam is the youngest
  - Else
    - Ajay is the youngest

Home Exercise #2

A certain grade of steel is graded according to the following conditions:

(i) Hardness must be greater than 50

(ii) Carbon content must be less than 0.7

(iii) Tensile strength must be greater than 5600

The grades are as follows:

- Grade is 10 if all three conditions are met
- Grade is 9 if conditions (i) and (ii) are met
- Grade is 8 if conditions (ii) and (iii) are met
- Grade is 7 if conditions (i) and (iii) are met
- Grade is 6 if only one condition is met
- Grade is 5 if none of the conditions are met

Write a program, which will require the user to give values of hardness, carbon content and tensile strength of the steel under consideration and output the grade of the steel.