

Conditional Statements (4-8-2025)

CODE: CSA0238

NAME: LOSHINI.L

REG NO: 192571015

1. Write a program to check if a number is positive, negative, or zero.

IPO

- Input Get a value as input, say n
- Process Check if the number is greater than, less than or equal to zero
n > 0 print (positive)
n < 0 print (negative)
n = 0 print (zero)
by using else if.
- Output The output is to print whether the given number is positive or negative or zero

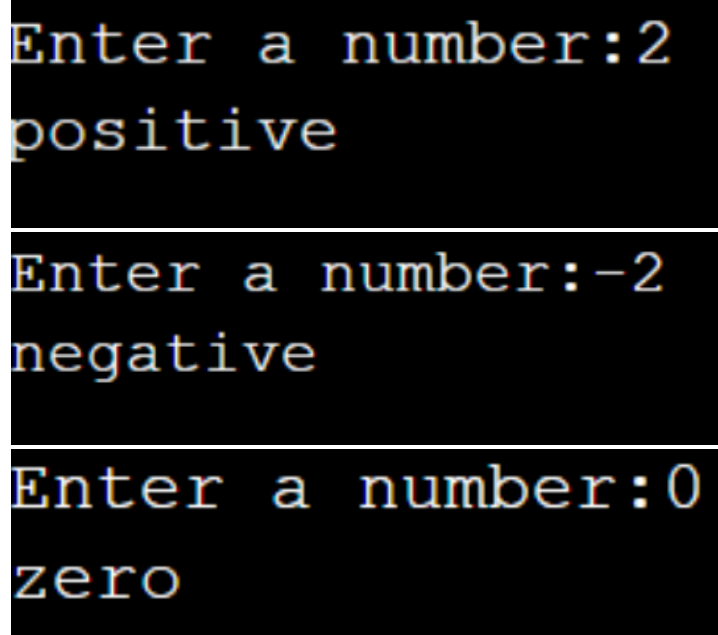
Program

```
#include<stdio.h>

void main()
{
    int n;
    printf("Enter a number:");
    scanf("%d",&n);
    if(n>0)
    {
        printf("positive");
```

```
}  
else if(n<0)  
{  
    printf("negative");  
}  
else  
{  
    printf("zero");  
}  
}
```

Output



The output consists of three separate screenshots of a terminal window. Each screenshot shows a prompt 'Enter a number:' followed by a user input and the program's response. The first screenshot shows input '2' and output 'positive'. The second screenshot shows input '-2' and output 'negative'. The third screenshot shows input '0' and output 'zero'.

```
Enter a number:2  
positive  
  
Enter a number:-2  
negative  
  
Enter a number:0  
zero
```

2. Write a program to find the largest among three numbers.

IPO

- Input
Get 3 values as input, say a, b, c
- Process
Compare the three numbers to determine the largest one
- Output
The output is to print the largest number

Program

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

```
void main()
```

```
{
```

```
    int a,b,c;
```

```
    printf("Enter three numbers: ");
```

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

```
    if(a>=b && a>=c)
```

```
    {
```

```
        printf("The largest number is:%d\n",a);
```

```
    }
```

```
    else if (b >= a && b >= c)
```

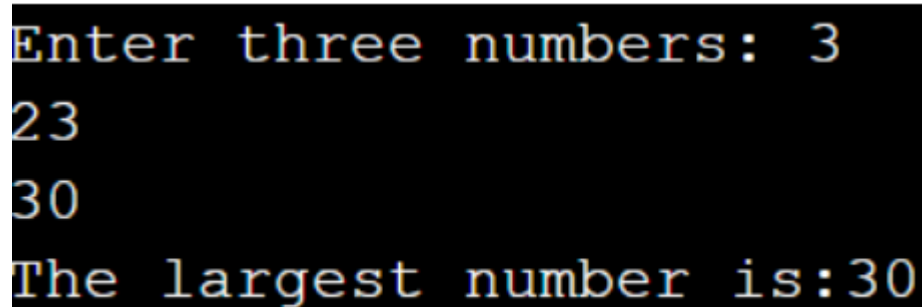
```
    {
```

```
        printf("The largest number is:%d\n",b);
```

```
    }
```

```
else
{
    printf("The largest number is:%d\n",c);
}
}
```

Output



```
Enter three numbers: 3
23
30
The largest number is:30
```

3. Write a program to check if a year is a leap year.

IPO

- Input: Get 1 value as input, say n
- Process: Check whether the given number $n \% 4 == 0$
Then it's a leap year else it's not a leap year
- Output: The output is to print whether the given number is leap year or not

Program

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

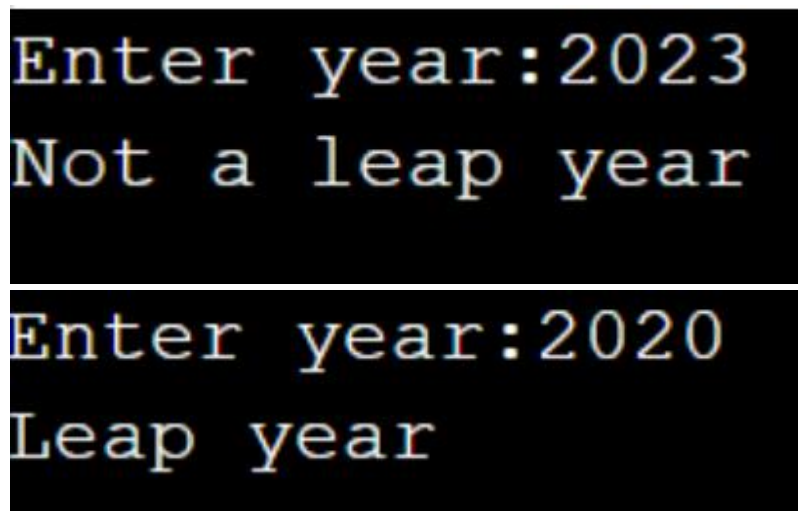
```
void main()
```

```
{
```

```
    int year;
```

```
printf("Enter year:");  
scanf("%d",&year);  
  
if(year%4==0)  
{  
    printf("Leap year");  
}  
else  
{  
    printf("Not a leap year");  
}  
}
```

Output



Enter year:2023
Not a leap year

Enter year:2020
Leap year

4. Write a program to check whether a character is a vowel or consonant.

IPO

- Input
Get 1 value as input, say ch
- Process
Check whether the character is one of
'a', 'e', 'i', 'o', 'u'
- Output
The output is to print whether the character is vowel or
a consonant

Program

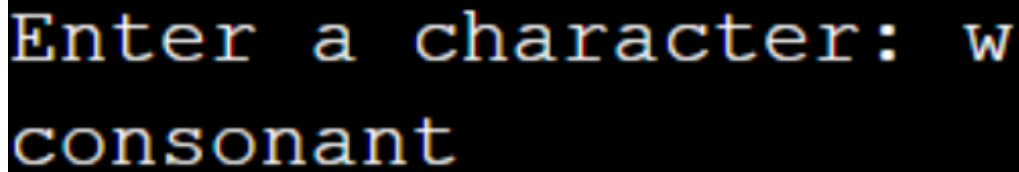
```
#include<stdio.h>

void main()
{
    char ch;
    printf("Enter a character: ");
    scanf("%c",&ch);

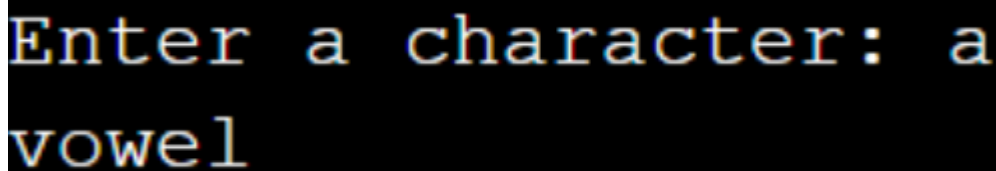
    if (ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' ||
        ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')
    {
        printf("vowel");
    }
    else
    {
```

```
printf("consonant");  
}
```

Output



```
Enter a character: w  
consonant
```



```
Enter a character: a  
vowel
```

5. Write a program to assign grades based on marks

IPO

- Input:
Get the marks as input (say n)
- Process:
Use if...else if statements to check the range of marks and assign the corresponding grade:
If $n \geq 90$ Print Grade A
Else if $n > 80$ Print Grade B
Else if $n > 70$ Print Grade C
Else if $n > 60$ Print Grade D
Else if $n < 60$ Print Grade F
Else Print Invalid grade

- Output:

Print the grade based on the entered marks

Program

```
#include<stdio.h>
void main()
{
    int n;
    printf("Enter marks: ");
    scanf("%d",&n);

    if(n>=90)
    {
        printf("Grade A");
    }
    else if(n>80)
    {
        printf("Grade B");
    }
    else if(n>70)
    {
        printf("Grade C");
    }
    else if(n>60)
    {
        printf("Grade D");
    }
    else if(n<60)
    {
        printf("Grade F");
    }
}
```



```

    }
    else
    {
        printf("Invalid grade");
    }
}

```

Output

```

Enter marks: 90
Grade A

```

6. Write a program to check whether a number is divisible by both 5 and 11

IPO

- Input:
Get a number say n as input
- Process: check if the number is divisible by 5 and 11 using modular operator,
If $n \% 5 == 0$ and $n \% 11 == 0$, then it is divisible by both
- Output:
Print whether the number is divisible by both 5 and 11 or not

Program

```

#include<stdio.h>

void main()
{
    int n;
    printf("Enter a number: ");

```

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

if(n%5==0 && n%11==0)
{
    printf("The number is divisible by 5 and 11 \n");
}
else
{
    printf("The number is not divisible by 5 and 11 \n");
}
}
```

Output

```
Enter a number: 55
The number is divisible by 5 and 11
```

```
Enter a number: 3
The number is not divisible by 5 and 11
```

7. Write a program to find the absolute value of a number.

IPO

- Input:
Get a number say n from the user.
- Process:
If $n < 0$, convert it to positive by $n = -n$.

- Output:

Print the absolute value of the number.

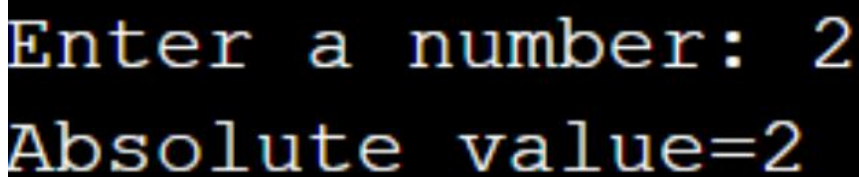
Program

```
#include<stdio.h>
void main()
{
    int n;
    printf("Enter a number: ");
    scanf("%d",&n);

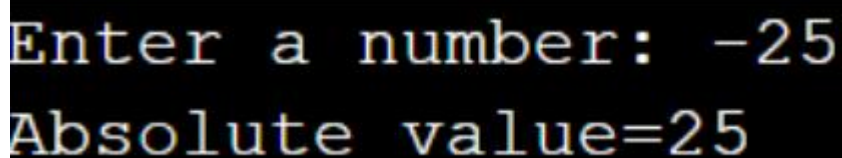
    if(n<0)
    {
        n=-n;
    }

    printf("Absolute value=%d\n",n);
}
```

Output



Enter a number: 2
Absolute value=2



Enter a number: -25
Absolute value=25

8. Write a menu-driven program to perform +, -, *, / operations.

IPO

- Input: get two numbers as input say a and b
- Process:
 - Case + Add the numbers $\rightarrow a + b$
 - Case - Subtract the numbers $\rightarrow a - b$
 - Case* Multiply the numbers $\rightarrow a * b$
 - Case % Check if $b \neq 0$, If true, divide the numbers $\rightarrow a / b$, Else print error
- Output:

The output is to print the result of the chosen operation (Addition, Subtraction, Multiplication, or Division), or to print an error

Program

```
#include <stdio.h>

void main()
{
    int choice;
    float a,b,result;
    scanf("%f%f",&a,&b);
    scanf("%d", &choice);
    switch(choice)
    {
```

case 1:

```
    result = a+b;  
    printf("Result=%.2f\n",result);  
    break;
```

case 2:

```
    result = a-b;  
    printf("Result=%.2f\n",result);  
    break;
```

case 3:

```
    result = a*b;  
    printf("Result=%.2f\n",result);  
    break;
```

case 4:

```
    if(b!=0)  
    {  
        result=a/b;  
        printf("Result=%.2f\n",result);  
    }  
    else  
    {  
        printf("Cannot divide by zero\n");  
    }
```

```

        break;
    default:
        printf("Invalid choice\n");
    }
}

```

Output

```

2
3
1
Result=5.00

```

```

2
3
2
Result=-1.00

```

```

2
3
3
Result=6.00

```

```

2
3
4
Result=0.67

```

9. Write a program to find roots of a quadratic equation.

IPO

Input:

Coefficients a, b, c of the quadratic equation

Process:

By using quadratic formula ,

$$d = b^2 - 4ac$$

If $d > 0$: roots are real and different

If $d == 0$: roots are real and equal

If $d < 0$: roots are complex (imaginary)

Output:

The output is to print the types of roots and values of roots using quadratic equation

Program

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

```
#include <math.h>
```

```
void main()
```

```
{
```

```
    float a,b,c,d,root1,root2;
```

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

```
    d=b*b-4*a*c;
```

```
    if(d>0)
```

```
    {
```

```
        root1=(-b+sqrt(d))/(2*a);
```

```
        root2=(-b-sqrt(d))/(2*a);
```

```

    printf("Roots are real and distinct:\n");
    printf("Root1=%.2f\n",root1);
    printf("Root2=%.2f\n",root2);
}
else if(d= 0)
{
    root1=root2=-b/(2*a);
    printf("Roots are real and equal:\n");
    printf("Root=%.2f\n",root1);
}
else
{
    float realPart =-b/(2*a);
    float imagPart =sqrt(-d)/(2*a);
    printf("Roots are complex and imaginary:\n");
    printf("Root1 = %.2f + %.2fi\n", realPart, imagPart);
    printf("Root2 = %.2f - %.2fi\n", realPart, imagPart);
}
}

```

Output


```
1
5
6
Roots are real and distinct:
Root1=-2.00
Root2=-3.00
```

10. Write a program to find the number of digits in a number.

IPO

- **Input:**

Get a value as input say n

Process:

If $n == 0$, count = 1

Otherwise, divide the number by 10 repeatedly using a loop

Increment count for each division until n becomes 0

- **Output:**

The output is to print the number of digits in a number

Program

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

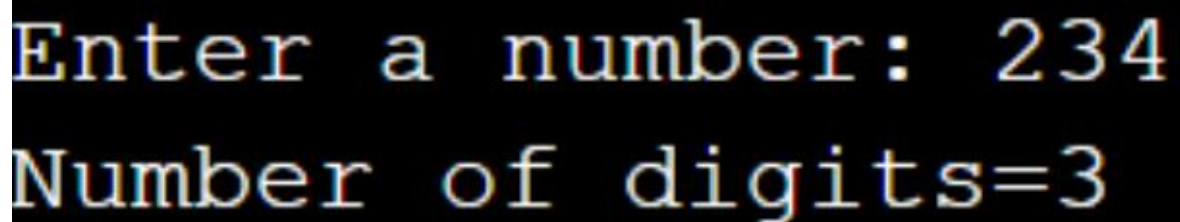
```
void main()
```

```
{
```

```
    int n,count=0;
```

```
printf("Enter a number: ");
scanf("%d",&n);
if(n==0)
{
    count=1;
}
else
{
    while(n!=0)
    {
        n=n/10;
        count++;
    }
}
printf("Number of digits=%d\n",count);
}
```

Output

A screenshot of a terminal window with a black background and white text. The first line shows the prompt 'Enter a number:' followed by the input '234'. The second line shows the output 'Number of digits=3'.

```
Enter a number: 234
Number of digits=3
```