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

IPO

**Input:**

* An integer number (a)

**Process:**

* Check the value of a:
  + If a > 0, then it's positive
  + If a < 0, then it's negative
  + If a == 0, then it's zero

**Output:**

* A message indicating whether the number is:
  + "The given number is positive"
  + "The given number is negative"

Code:

#include<stdio.h>

void main()

{

int a,i;

scanf("%d",&a);

if(a>0)

{

printf("the given number is positive");

}

else

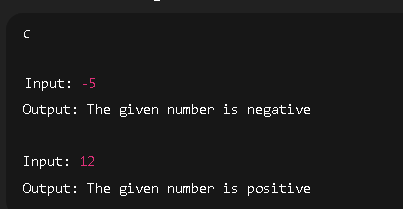
{

printf("the given number is negative");

}

}

OUTPUT



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

IPO

**Input:**

* Three integers a, b, c

**Process:**

* Compare all three values:
  + If a >= b and a >= c → A is greatest
  + Else if b >= a and b >= c → B is greatest

Code:

#include<stdio.h>

void main()

{

int a,b,c;

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

if(a>b)

{

printf("A is greater");

}

else if(b>c)

{

printf("B is greater");

}

else

{

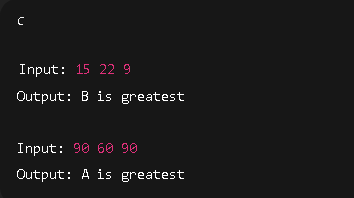
printf("C is Greater");

}

}

}

OUTPUT



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

IPO

**Input:**

* An integer a (year)

**Process:**

* If (a % 4 == 0 && a % 100 != 0) || (a % 400 == 0) → It's a leap year
* Else → It's not a leap year

**Output:**

* Print "its a leap year" or "its not a leap year"

CODE

#include<stdio.h>

void main()

{

int a;

scanf("%d",&a);

if((a%4==0 && a%100!=0)||(a%400==0))

{

printf("its a leap year");

}

else

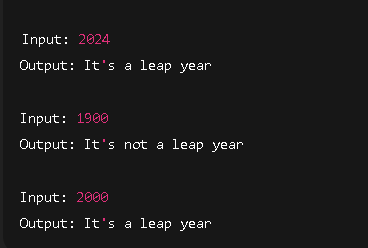
{

printf("its not a leap year");

}

}

OUTPUT



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

IPO

**Input:**

* A single character (ch)
* Check if ch is an alphabet
* If it's a vowel → Print "It is a vowel"
* Else if alphabet but not vowel → Print "It is a consonant"

#include <stdio.h>

void main()

{

char ch;

scanf("%c", &ch);

if ((ch>='A' && ch<='Z') || (ch>='a' && ch<='z'))

{

if (ch=='A' || ch=='E' || ch=='I' || ch =='O' || ch=='U' ||

ch=='a' || ch=='e' || ch=='i' || ch =='o' || ch=='u')

{

printf("It is a vowel");

}

else

{

printf("It is a consonant");

}

}

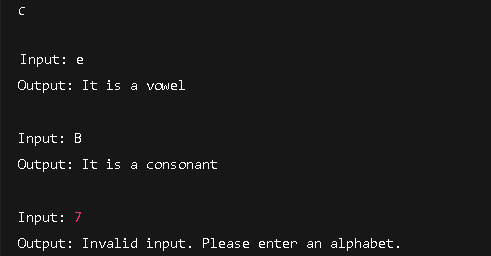
else

{

printf("Invalid input. Please enter an alphabet.");

}

OUTPUT



1. Write a program to assign grades based on marks.

IPO

**Input:**

* Integer a (marks)

**Process:**

* If a is out of valid range → invalid
* If a ≥ 95 → A grade
* Else if a ≥ 85 → B grade

Code

#include <stdio.h>

void main()

{

int a;

scanf("%d",&a);

if(a>=95)

{

printf("A grade");

}

else if(a>=85)

{

printf("B grade");

}

else

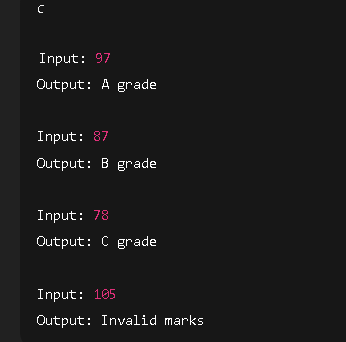
{

printf("C grade");

}

}

OUTPUT



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

**Input:**

* Integer a

**Process:**

* If a % 5 == 0 && a % 11 == 0 → divisible
* Else → not divisible

#include <stdio.h>

void main()

{

int a;

scanf("%d",&a);

if(a%5==0 && a%11==0)

{

printf("its is divisible");

}

else

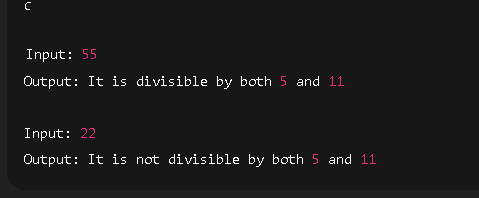
{

printf("its not divisible");

}

}

OUTPUT



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

IPO

**Input:**

* Integer a

**Process:**

* If a < 0, convert to positive by a = -a

**Output:**

* Print the absolute value

CODE

#include <stdio.h>

void main()

{

int a;

scanf("%d", &a);

if (a < 0)

{

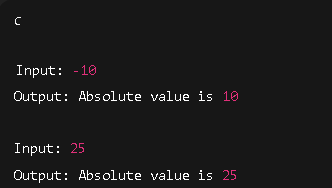
a = -a;

}

printf("Absolute value is %d", a);

}

OUTPUT



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

IPO

**Input:**

* Two numbers (a and b)
* User’s choice (1–4)

**Process:**

* Perform the selected arithmetic operation using switch

**Output:**

* Display result or error message
* CODE

#include <stdio.h>

void main()

{

int choice;

float a, b, result;

printf("Enter two numbers: ");

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

printf("\nMenu:\n");

printf("1. Addition (+)\n");

printf("2. Subtraction (-)\n");

printf("3. Multiplication (\*)\n");

printf("4. Division (/)\n");

printf("Enter your choice (1-4): ");

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("Error: Division by zero is not allowed.\n");

}

break;

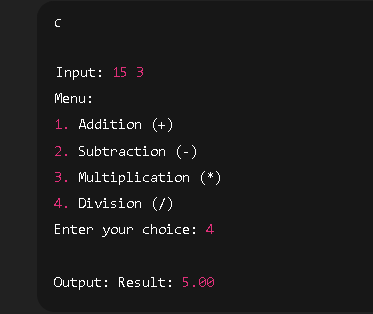
default:

printf("Invalid choice. Please enter a number between 1 and 4.\n");

}

}

OUTPUT



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

**Input:**

* Coefficients a, b, c

**Process:**

* Calculate discriminant: D = b² - 4ac
* Based on D:
  + D > 0 → real and distinct roots
  + D == 0 → real and equal roots
  + D < 0 → complex roots

**Output:**

* Displays the nature and values of the roots

CODE

#include <stdio.h>

#include <math.h>

void main()

{

float a, b, c, discriminant, root1, root2;

printf("Enter coefficients a, b and c: ");

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

discriminant = b \* b - 4 \* a \* c;

if (discriminant > 0)

{

root1 = (-b + sqrt(discriminant)) / (2 \* a);

root2 = (-b - sqrt(discriminant)) / (2 \* a);

printf("Roots are real and distinct:\n");

printf("Root1 = %.2f\n", root1);

printf("Root2 = %.2f\n", root2);

}

else if (discriminant == 0)

{

root1 = -b / (2 \* a);

printf("Roots are real and equal:\n");

printf("Root = %.2f\n", root1);

}

else

{

float realPart = -b / (2 \* a);

float imagPart = sqrt(-discriminant) / (2 \* a);

printf("Roots are complex and imaginary:\n");

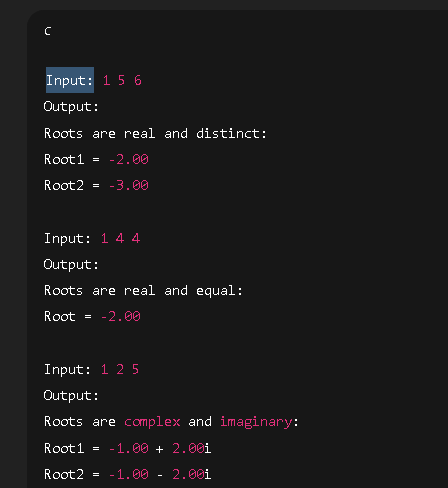
printf("Root1 = %.2f + %.2fi\n", realPart, imagPart);

printf("Root2 = %.2f - %.2fi\n", realPart, imagPart);

}

}

OUTPUT



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

IPO

Input: 0 → Output: Number of digits = 1

Input: 1234 → Output: Number of digits = 4

Input: -98 → Output: Number of digits = 2

CODE

#include <stdio.h>

void main()

{

int num, count=0;

scanf("%d", &num);

if (num == 0)

{

printf("its not a number");

}

while (num != 0)

{

num = num/10;

count++;

}

printf("Number of digits = %d\n", count);

}

OUTPUT

