

**Que1. WAP in C++ "hello, world!" program.**

**Ans:- #include<iostream.h>**

**#include<conio.h>**

**Void main()**

**{**

**Clrscr();**

**Cout<<"hello, world!";**

**Getch();**

**}**

**Que2. WAP in C++ Program to Calculate Sum of Natural Numbers.**

**Ans:- #include <iostream.h>**

**Using namespace std;**

**Int main()**

**{**

**Int n, sum = 0;**

**Cout << "Enter a natural number: ";**

**Cin >> n;**

**For (int i = 1; i <= n; i++)**

**{**

**Sum += i;**

**}**

**Cout << "Sum of natural numbers from 1 to " << n << " is: " << sum << endl;**

**Return 0;**

**}**

**Que3. WAP in C++ program to swap two numbers**

**Ans:- #include <iostream>**

**Using namespace std;**

```

Int main()
{
Int num1, num2, temp;
Cout << "Enter first number: ";
Cin >> num1;
Cout << "Enter second number: ";
Cin >> num2;
Cout << "Before swapping: " << endl;
Cout << "Number 1: " << num1 << endl;
Cout << "Number 2: " << num2 << endl;

// Swap numbers
Temp = num1;
Num1 = num2;
Num2 = temp;

Cout << "After swapping: " << endl;
Cout << "Number 1: " << num1 << endl;
Cout << "Number 2: " << num2 << endl;
Return 0;
}

```

Que4. WAP to demonstrate else if ladder.

Ans:- #include <iostream>

Using namespace std;

Int main()

{

Int marks;

Cout << "Enter your marks: ";

```
Cin >> marks;
If (marks >= 90)
{
    Cout << "Grade: A" << endl;
}
else if (marks >= 80)
{
    Cout << "Grade: B" << endl;
}
else if (marks >= 70)
{
    Cout << "Grade: C" << endl;
}
else if (marks >= 60)
{
    Cout << "Grade: D" << endl;
}
else
{
    Cout << "Grade: F" << endl;
}
Return 0;
}
```

**Que5. WAP to demonstrate while loop**

**Ans:- #include<iostream.h>**

**#include<conio.h>**

**Void main()**

**{**

```
Int a;
Clrscr();
A=1;
While(a<=10)
{
Cout<<a<<"\n";
A++;
}
Getch();
}
```

**Que6. WAP to demonstrate for loop.**

**Ans:- #include<iostream.h>**

**#include<conio.h>**

**Void main()**

```
{
Int a;
Clrscr();
For(a=1; a<=10; a++)
{
Cout<<a<<"\n";
}
Getch();
}
```

**Output:- 1,2,3,4,5,6,7,8,9,10**

**Que7. WAP to demonstrate goto and lable.**

**Ans:- #include <iostream>**

**Using namespace std;**

```

Int main()
{
    Int i = 1;
    Label:
    Cout << i << " ";
    I++;
    If (i <= 10)
    {
        Goto label;
    }
    Cout << endl;
    Return 0;
}

```

Output:- 1,2,3,4,5,6,7,8,9,10

Que8. WAP to demonstrate array creation, insertion elements and print.

Ans:- #include <iostream>

Using namespace std;

Int main()

{

// Array creation

Int arr[5];

// Insertion elements

Cout << "Enter 5 elements: " << endl;

For (int i = 0; i < 5; i++)

{

Cin >> arr[i];

```
}  
  
// Print array elements  
Cout << "Array elements: " << endl;  
For (int i = 0; i < 5; i++)  
{  
    Cout << arr[i] << " ";  
}  
Cout << endl;  
Return 0;  
}
```

**Output:- Enter 5 elements:**

10

20

30

40

50

**Array elements: 10 20 30 40 50**

**Que.9. WAP in C++ program to check whether a number is even or odd.**

**Ans:- #include <iostream>**

**Using namespace std;**

**Int main()**

**{**

**Int num;**

**Cout << "Enter a number: ";**

**Cin >> num;**

**If (num % 2 == 0)**

**{**

```
    Cout << num << " is even." << endl;
}
else
{
    Cout << num << " is odd." << endl;
}
Return 0;
}
```

Output:- Enter a number: 10

10 is even.

Que10. WAP in C++ program to display fibonacci sequence using recursion.

Ans:- #include <iostream>

Using namespace std;

Int fibonacci(int n)

{

If (n == 0 || n == 1)

{

Return n;

}

else

{

Return fibonacci(n-1) + fibonacci(n-2);

}

}

Int main()

{

Int n;

```

Cout << "Enter the number of terms: ";
Cin >> n;
Cout << "Fibonacci sequence: ";
For (int i = 0; i < n; i++)
{
    Cout << fibonacci(i) << " ";
}
Cout << endl;
Return 0;
}

```

Output:- Enter the number of terms: 10  
Fibonacci sequence: 0 1 1 2 3 5 8 13 21 34

Que11. WAP in C++ program to check whether a number is palindrome or not

```

Ans:- #include <iostream>
Using namespace std;
Int main()
{
    Int num, reverse = 0, remainder, original;
    Cout << "Enter a number: ";
    Cin >> num;

    Original = num;
    While (num != 0)
    {
        Remainder = num % 10;
        Reverse = reverse * 10 + remainder;
        Num /= 10;
    }
}

```



```

    }
    If (original == reverse)
    {
        Cout << original << " is a palindrome number." << endl;
    }
    else
    {
        Cout << original << " is not a palindrome number." << endl;
    }
    Return 0;
}

```

Output:- Enter a number: 121

121 is a palindrome number.

Que12. WAP in C++ program to check armstrong number.

Ans:- #include <iostream>

Using namespace std;

Int main()

```

{
    Int num, original, remainder, result = 0, n = 0;

```

```

    Cout << "Enter a number: ";

```

```

    Cin >> num;

```

```

    Original = num;

```

```

    While (original != 0)

```

```

{
    Original /= 10;

```

```

    N++;
}

Original = num;
While (original != 0)
{
    Remainder = original % 10;
    Result += pow(remainder, n);
    Original /= 10;
}

If (result == num)
{
    Cout << num << " is an Armstrong number." << endl;
}
else
{
    Cout << num << " is not an Armstrong number." << endl;
}
Return 0;
}

```

**Output:- Enter a number: 153**

**153 is an Armstrong number**

**Que13. WAP in C++ program to make a simple calculator using switch...case**

**Ans:- #include <iostream>**

**Using namespace std;**

```
Int main()
{
    Int choice, num1, num2, result;

    Cout << "Simple Calculator" << endl;
    Cout << "1. Addition" << endl;
    Cout << "2. Subtraction" << endl;
    Cout << "3. Multiplication" << endl;
    Cout << "4. Division" << endl;
    Cout << "Enter your choice: ";
    Cin >> choice;

    Cout << "Enter first number: ";
    Cin >> num1;

    Cout << "Enter second number: ";
    Cin >> num2;

    Switch (choice)
    {
        Case 1:
            Result = num1 + num2;
            Cout << "Result: " << result << endl;
            Break;
        Case 2:
            Result = num1 - num2;
            Cout << "Result: " << result << endl;
            Break;
        Case 3:
```

```
Result = num1 * num2;

Cout << "Result: " << result << endl;

Break;

Case 4:

If (num2 != 0)

{

    Result = num1 / num2;

    Cout << "Result: " << result << endl;

}

else

{

    Cout << "Error: Division by zero!" << endl;

}

Break;

Default:

Cout << "Invalid choice!" << endl;

Break;

}

Return 0;

}
```

**Que14. WAP in C++ to demonstrate the constructor and destructor.**

**Ans:- #include <iostream>**

**Using namespace std;**

**Class Student**

```
{

    Private:

    String name;
```

```
Int age;
```

```
Public:
```

```
// Constructor
```

```
Student(string n, int a)
```

```
{
```

```
    Name = n;
```

```
    Age = a;
```

```
    Cout << "Constructor called." << endl;
```

```
}
```

```
// Destructor
```

```
~Student()
```

```
{
```

```
    Cout << "Destructor called." << endl;
```

```
}
```

```
// Method to display student details
```

```
Void displayDetails()
```

```
{
```

```
    Cout << "Name: " << name << endl;
```

```
    Cout << "Age: " << age << endl;
```

```
}
```

```
};
```

```
Int main()
```

```
{
```

```
    Student s1("John Doe", 20);
```

```
    S1.displayDetails();
```

```
Return 0;
```

```
}
```

**Output:- Constructor called.**

**Name: John Doe**

**Age: 20**

**Destructor called.**

**Que15. WAP in C++ to demonstrate the function overloading.**

**Ans:- #include <iostream>**

**Using namespace std;**

**Class Calculator**

```
{
```

```
    Public:
```

```
        Int add(int a, int b)
```

```
{
```

```
    Return a + b;
```

```
}
```

```
        Double add(double a, double b)
```

```
{
```

```
    Return a + b;
```

```
}
```

```
        Int add(int a, int b, int c)
```

```
{
```

```
    Return a + b + c;
```

```
}
```

```
};
```

```
Int main()
{
    Calculator calc;

    Cout << "Addition of two integers: " << calc.add(10, 20) << endl;
    Cout << "Addition of two doubles: " << calc.add(10.5, 20.7) << endl;
    Cout << "Addition of three integers: " << calc.add(10, 20, 30) << endl;
    Return 0;
}
```

**Output:**

**Addition of two integers: 30**

**Addition of two doubles: 31.2**

**Addition of three integers: 60**

**Que16. WAP in C++ to demonstrate the inheritance.**

**Ans:- #include <iostream>**

**#include <string>**

**Using namespace std;**

**// Base class**

**Class Vehicle**

**{**

**Protected:**

**String brand;**

**String model;**

**Public:**

```
Vehicle(string b, string m)
{
    Brand = b;
    Model = m;
}

Void displayDetails()
{
    Cout << "Brand: " << brand << endl;
    Cout << "Model: " << model << endl;
}
};

// Derived class
Class Car : public Vehicle
{
    Private:
        Int numDoors;

    Public:
        Car(string b, string m, int n) : Vehicle(b, m)
        {
            numDoors = n;
        }

    Void displayCarDetails()
    {
        displayDetails();
        cout << "Number of Doors: " << numDoors << endl;
    }
}
```



```

    }
};

Int main()
{
    Car myCar("Toyota", "Corolla", 4);
    myCar.displayCarDetails();

    return 0;
}

```

**Output:**

**Brand: Toyota**

**Model: Corolla**

**Number of Doors: 4**

**Que17. WAP in C++ Full Pyramid of \***

```

          *
         **
        ***
       ****
      *****
     *

```

**Ans:- #include <iostream>**

**Using namespace std;**

**Int main()**

**{**

**Int rows;**

```
Cout << "Enter the number of rows: ";
Cin >> rows;

For (int i = 0; i < rows; i++)
{
    For (int j = 0; j < rows - i - 1; j++)
    {
        Cout << " ";
    }
    For (int k = 0; k <= i; k++)
    {
        Cout << "* ";
    }
    Cout << endl;
}

Return 0;
}
```

**Output:**

Enter the number of rows: 5

```
*
**
***
****
*****
```

