XI STD COMPUTER SCIENCE C++Programs 2024 - 2025

Sl.No	Program	Output

1. School Address	#include <iostream>
	using namespace std;
	int main()
	{
	cout << "\n GOVT. HR.SEC.SCHOOL ";
	cout << "\n MUKHASAPARUR ";
	cout << "\n CUDDALORE DT ";
	cout << "\n PIN 606001" ;
	}
		GOVT. HR.SEC.SCHOOL
		MUKHASAPARUR
		CUDDALORE DT
		PIN 606001

2. வட்டத்தின் பரப்பளவை கண்டறியும் C++ நிரல்.(area of circle)	#include <iostream>
	using namespace std;
	int main()
	{
	int radius;
	float area;
	cout << "\n Enter Radius: ";
	cin >> radius;
	area = 3.14 * radius * radius;
	cout << "\n The area of circle = " << area;
	}
		Enter Radius: 8
		The area of circle = 200.96

3. கூட்டுத் தொகை (SUM of 3 marks)	#include <iostream>
	using namespace std;
	int main()
	{
	int m1, m2, m3, sum;
	cout << "\n Enter Mark 1: ";
	cin >> m1;
	cout << "\n Enter Mark 2: ";
	cin >> m2;
	cout << "\n Enter Mark 3: ";
	cin >> m3;
	sum = m1 + m2 + m3;
	cout << "\n The sum = " << sum;
	}
		Enter Mark 1: 45
		Enter Mark 2: 50
		Enter Mark 3: 55
		The sum = 150

4. if	#include <iostream>
	using namespace std;
	int main()
	{
	int age;
	cout << "\n Enter your age: ";
	cin >> age;
	if (age >= 18)
	cout<< "\n You are eligible for voting ....";
	cout << "This statement is always executed.";
	return 0;
	}
		Enter your age: 21
		You are eligible for voting....This statement is always executed.
		
		Enter your age: 17
		This statement is always executed.

5. if..else	#include <iostream>
	using namespace std;
	int main()
	{
	int num, rem;
	cout<< "\n Enter a number: ";
	cin>>num;
	rem = num % 2;
	if (rem==0)
	cout<< "\n The given number" <<num<< " is Even";
	else
	cout<< "\n The given number "<<num<< " is Odd";
	return 0;
	}
		Enter a number: 54
		The given number54 is Even
		
		Enter a number: 31
		The given number 31 is Odd

6. if -else-if	#include <iostream>
	using namespace std;
	int main()
	{
	int marks;
	cout<<" Enter the Marks :";
	cin>>marks;
	if(marks >= 60)
	cout<< "Your grade is 1st class !!" <<endl;
	else if(marks >= 50 && marks < 60)
	 cout<< "your grade is 2nd class !!" <<endl;
	 else if(marks >= 40 && marks < 50)
	 cout<< "your grade is 3rd class !!" <<endl;
	else
	cout<< "You are fail !!" <<endl;
	return 0;
	}
		Enter the Marks :70
		Your grade is 1st class !!
		
		Enter the Marks :59
		your grade is 2nd class !!
		
		Enter the Marks :49
		your grade is 3rd class !!
		
		Enter the Marks :39
		You are fail !!

7. நிபந்தனை செயற்குறி (Conditional operator)	#include <iostream>
	using namespace std;
	int main()
	{
	int a, b, largest;
	cout << "\n Enter any two numbers: ";
	cin >> a >> b;
	 largest = (a>b)? a : b;
	cout << "\n Largest number : " << largest;
	return 0;
	}
		Enter any two numbers: 23
		45
		
		 Largest number : 45

8. switch	#include <iostream>
	using namespace std;
	int main()
	{
	int num;
	cout << "\n Enter week day number: ";
	cin >> num;
	switch (num)
	{
	 case 1 : cout << "\n Sunday"; break;
	 case 2 : cout << "\n Monday"; break;
	 case 3 : cout << "\n Tuesday"; break;
	 case 4 : cout << "\n Wednesday"; break;
	 case 5 : cout << "\n Thursday"; break;
	 case 6 : cout << "\n Friday"; break;
	 case 7 : cout << "\n Saturday"; break;
	 default: cout << "\n Wrong input....";
	}
	}
		Enter week day number: 1
		 Sunday
		
		Enter week day number: 2
		 Monday
		
		Enter week day number: 3
		 Tuesday
		
		Enter week day number: 4
		 Wednesday

9. for	#include <iostream>
	using namespace std;
	int main ()
	{
	int i;
	for(i = 1; i < 10; i ++)
	cout<<i<<"\t";
	return 0;
	}
		1 2 3 4 5 6
		7 8 9 10

10. while	#include <iostream>
	using namespace std;
	int main ()
	{
	int i=1;
	while(i<=10)
	{
	cout<<i<<"\t";
	i+=1;
	}
	return 0;
	}
		1 2 3 4 5 6
		7 8 9 10

11. do…while	#include <iostream>
	using namespace std;
	int main ()
	{
	int i=1;
	do
	{
	cout<<i<<"\t";
	i+=1;
	}while(i<=10);
	return 0;
	}
		1 2 3 4 5 6
		7 8 9 10

12. while மடக்கை பயன்படுத்தி 1 முதல் 10 வரை உள்ள எண்களின் தொடர் கூட்டலை வெளியிடும் நிரல் எழுதுக(The sum of 1 to 10 using while loop)	#include <iostream>
	using namespace std;
	int main ()
	{
	int i=1,sum=0;
	while(i<=10)
	{
	sum=sum+i;
	i++;
	}
	cout<<"The sum of 1 to 10 is "<<sum;
	return 0;
	}
		The sum of 1 to 10 is 55

13. do-while() மடக்கினைப் பயன்படுத்தி 10 முதல் 1 வரை உள்ள எண்களை வெளியிட நிரல் எழுதுக(display the series 10,9,8,.....,1 using do.while loop)	#include <iostream>
	using namespace std;
	int main ()
	{
	int n = 10;
	do
	{
	cout<<n<<", ";
	n--;
	}while (n>0) ;
	}
		10, 9, 8, 7, 6, 5, 4, 3, 2, 1,

14.go to	# include <iostream>
	using namespace std;
	int main()
	{
	int n=1;
	jump:
	{
	 if(n<10)
	 {
	cout<<n<<'\t';
	n+=2;
	goto jump;
	 }
	 return 0;
	}
	}
		1 3 5 7 9

15.break	# include <iostream>
	using namespace std;
	int main ()
	{
	int count = 1;
	do
	{
	cout<< "Count : " << count <<endl;
	if(count > 3)
		Count : 1
		Count : 2
		Count : 3
		Count : 4

	{
	break;
	}
	count ++;
	}while(count < 20);
	return 0;
	}	

16. continue	#include <iostream>
	using namespace std;
	int main()
	{
	for (int i = 1; i<= 10; i++)
	{
	if (i == 6)
	continue;
	else
	cout<<i<< " ";
	}
	return 0;
	}
		1 2 3 4 5 7 8 9 10

17. ஒரு முழு எண்ணின் தொடர் பெருக்கல(Factorial) கணக்கிடும் C++ நிரல் ஒன்றை எழுதுக	#include<iostream>
	using namespace std;
	int main() {
	   int n;
	   long factorial = 1.0;
	
	   cout << "Enter a positive integer: ";
	   cin >> n;
	
	   if (n < 0)
	      cout << "Error! Factorial of a negative number doesn't exist.";
	   else {
	      for(int i = 1; i <= n; ++i) {
	         factorial *= i;
	      }
	      cout << "Factorial of " << n << " = " << factorial;
	   }
	
	   return 0;
	}
		Enter a positive integer: 5
		Factorial of 5 = 120

18. ஃபிபோனாசி எண் வரிசை(Fibonacci series) (0 1 1 2 3 5 8 …….) கண்டறியும் C++ நிரல் ஒன்றை எழுதுக.	#include <iostream>
	using namespace std;
	int main() {
	   int n, t1 = 0, t2 = 1,t3 = 0;
	   cout << "Enter the number of terms: ";
	   cin >> n;
	   cout << "Fibonacci Series: ";
	   cout << t1 << ", ";
	   cout << t2 << ", ";
	   for (int i = 1; i <= n-2; ++i)
	   {
	         t3= t1 + t2;
	      t1 = t2;
	      t2 = t3;
	     cout << t3 << ", ";
	   }
	   return 0;
	}
		Enter the number of terms: 10
		Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34,

19. A
A B
A B C
A B C D
A B C D E
A B C D E F
	#include <iostream>
	#include <conio.h>
	using namespace std;
	int main()
	{
	int i,j,rows;
	rows=6;
	for(i=1; i<=rows; i++){
	for(j=1; j<=i; j++){
	cout<<((char)(j+64));
	}
	cout<< endl;
	}
	getch();
	return 0;
	}
		A
		A B
		A B C
		A B C D
		A B C D E
		A B C D E F

20. 5 4 3 2 1
5 4 3 2
5 4 3
5 4
5
	#include<iostream>
	using namespace std;
	int main ()
	{
	int i,j;
	for(i=1; i<=5; i++)
	{
	for(j=5; j>=i; j--)
	{
	cout<<j<<" ";
	}
	cout<<"\n";
	}
	return 0;
	}
		5 4 3 2 1
		5 4 3 2
		5 4 3
		5 4
		5

21. கொடுக்கப்பட்ட எண்ணின் பெருக்கல் வாய்ப்பாட்டை அச்சிடும் C++ நிரல் ஒன்றை எழுதுக **(Multiflication Table)**	**#include<iostream>**
	using namespace std;
	int main( )
	{
	int n;
	cout<<"Enter the Table number to print : ";
	cin>>n;
	for(int i=1;i<=10;i++)
	cout<<i<<"x"<<n<<"="<<i*n<<endl;
	return 0;
	}
		Enter the Table number to print : 7
		1x7=7
		2x7=14
		3x7=21
		4x7=28
		5x7=35
		6x7=42
		7x7=49
		8x7=56
		9x7=63
		10x7=70

22. பின்வரும் எண் தொடரை அச்சிடுவதற்கான நிரல்களை எழுதுக.
(a) 1 4 7 10…… 40
	#include<iostream>
	using namespace std;
	int main( )
	{
	for (int i=1; i<=40 ; i+=3)
	cout << i<<',' ;
	return 0; }
		1,4,7,10,13,16,19,22,25,28,31,34, 37,40,

23.செயற்கூறு **(funtion) example**	**#include <iostream>**
	using namespace std;
	int sum (int x, int y)
	{ return (x + y); }
	int main ()
	{
	int a,b ;
	cout<<"\n Enter Number 1:";
	cin>>a;
	cout<<"\n Enter Number 2:";
	cin >>b;
	cout<<"\n The sum = "<<sum (a,b); }
		Enter Number 1:56
		Enter Number 2:45
		The sum = 101

24. முன்னியல்புச் செயலுருபுக்கள் (Default Argument)	#include <iostream>
	using namespace std;
	double area(const double r,const double pi=3.14)
	{
	return(pi*r*r);
	}
	int main ()
	{
	double rad,res;
	cout<<"\nEnter Radius :";
	cin>>rad;
	res=area(rad);
	cout << "\nThe Area of Circle ="<<res;
	return 0;
	}
		Enter Radius :8
		The Area of Circle =200.96

25.inline function	#include <iostream>
	using namespace std;
	int sum (int x, int y)
	{
	return (x + y);
	}
	int main ()
	{
	int a,b ;
	cout<<"\n Enter Number 1:";
	cin>>a;
	cout<<"\n Enter Number 2:";
	cin >>b;
	cout<<"\n The sum = "<<sum (a,b);
	}
		Enter Number 1:56
		Enter Number 2:45
		The sum = 101

26. வரையெல்லை தெளிவுபடுத்தும் செயற்குறி (Scope resolution operator)	#include<iostream>
	using namespace std;
	int x=45; // Global Variable x
	int main()
	{
	int x = 10; // Local Variable x
	cout << "\nValue of global x is " << ::x;
	cout << "\n\nValue of local x is " << x;
	return 0;
	}
		Value of global x is 45
		Value of local x is 10

27. C++ மொழியில் 10 மதிப்புகளை உள்ளீடாக பெற்று அதில் ஒற்றை எண்களின் எண்ணிக்க மற்றும் இரட்டை எண்களின் எண்ணிக்கையை காண்பதற்கான நிரல் (Count odd numbers &Even numbers).	#include <iostream>
	using namespace std;
	int main()
	{
	int num[10], even=0, odd=0;
	for (int i=0; i<10; i++)
	{
	cout<< "\n Enter Number " << i+1 <<"= ";
	cin>>num[i];
	if (num[i] % 2 == 0)
	++even;
	else
	++odd;
	}
	cout << "\n There are "<< even <<" Even Numbers";
	cout << "\n There are "<< odd <<" Odd Numbers";
	}
		Enter Number 1= 50
		Enter Number 2= 51
		Enter Number 3= 52
		Enter Number 4= 53
		Enter Number 5= 54
		Enter Number 6= 55
		Enter Number 7= 56
		Enter Number 8= 57
		Enter Number 9= 58
		Enter Number 10= 59
		There are 5 Even Numbers
		There are 5 Odd Numbers

28. உள்ளமை பொருளின் பயன்பாட்டை விளக்கும் நிரல் (The use of Local & global Object)	#include <iostream>
	using namespace std;
	class add
	{
	int a,b;
	public:
	int sum;
	void getdata()
	class add
	{
	int a,b;
	public:
	int sum;
	void getdata()
	{
	a=5;
	b=10;
	sum = a+b;
	}
	} a1;
	add a2;
	int main()
	{
	add a3;
	a1.getdata();
	a2.getdata();
	a3.getdata();
	cout<<a1.sum;
	cout<<a2.sum;
	cout<<a3.sum;
	return 0;
	}
		151515

29. பொருளுக்கான நினைவக இட ஒதுக்கீட்டை விளக்கும் நிரல்(memory allocation of objects)	# include <iostream>
	using namespace std;
	class product
	{
	int code, quantity;
	float price;
	public:
	void assignData();
	void Print();
	};
	int main()
	{
	product p1, p2;
	cout<<"\n Memory allocation for object p1 "
	<<sizeof(p1);
	cout<<"\n Memory allocation for object p2 "
	<<sizeof(p2);
	return 0;
	}
		Memory allocation for object p1 12
		Memory allocation for object p2 12

30. பொருள் இயங்குநிலையில் தொடங்குதல் (Dynamic initialization of objects)	#include<iostream>
	using namespace std;
	class X
	 {
	int n;
	 float avg;
	 public:
	X(int p,float q)
	{ n=p;
	avg=q;
	 }
		Enter the Roll Number 45
		Enter the Average 55.5
		 Roll number:- 45
		Average :- 55.5

```
void disp()
{
cout<<"\n Roll number:- " <<n;
cout<<"\n Average :- "<<avg;
}
 };
int main()
{
int a ; float b;
cout<<"\nEnter the Roll Number";
cin>>a;
cout<<"\nEnter the Average";
cin>>b;
 X x(a,b); // dynamic initialization
x.disp();
 return 0;
}
```

31. ஆக்கிகளின் வகைகள் (Types of constructor)

```
#include<iostream>
using namespace std;
class Data
 {
 int i, j;
 public:
 int k;
 Data()
 {
 cout<<"\n Non Parametrerized
constructor";
 i=0;
 j=0;
 }
 Data(int a,int b)
{
 cout<<"\n Parametrerized constructor";
 i=a;
 j=b;
 }
 Data(Data &a)
 {
 cout<<"\n Copy constructor";
 i=a.i;
 j=a.j;
 }
void display() //member function
 {
 cout<<"\n"<< i<<"\t"<<j;
 }
 };
int main()
{
Data d1,d2(10,20),d3(d2);
 d1.display();
 d2.display();
 d3.display();
 return 0;
}
```

```
Non Parametrerized constructor
 Parametrerized constructor
 Copy constructor
0      0
10     20
10     20
```

32. ஆக்கி மற்றும் அழிப்பி EXAMPLE (example of constructor & Destructor)	#include<iostream>
	using namespace std;
	class simple
	{
	private:
	int a, b;
	public:
	simple()
	{
	a= 0 ;
	b= 0;
	cout<< "\n Constructor of class-simple ";
	}
	void getdata()
	{
	cout<<"\n Enter values for a and b ";
	cin>>a>>b;
	}
	void putdata()
	{
	cout<<"\nThe two integers are .. ";
	cout<<a<<"\t"<<b<<endl;
	cout<<"\n The sum = "<<a+b;
	}
	~simple()
	{
	cout<<"\n Destructor is executed ";
	}
	};
	int main()
	{
	simple s;
	s.getdata();
	s.putdata();
	return 0;
	}
		Constructor of class-simple
		Enter values for a and b 10 15
		The two integers are .. 10 15
		The sum = 25
		Destructor is executed

33. செயற்கூறு பணிமிகுப்பு (Function overloading)	#include <iostream>
	using namespace std;
	void print(int i)
	{
	cout<< " It is integer " << i <<endl;
	}
	void print(double f)
	{
	cout<< " It is float " << f <<endl;}
	void print(string c)
	{
	cout<< " It is string " << c <<endl;
	}
	int main()
	{
	print(10);
	print(10.10);
	print("Ten");
	return 0;
	}
		It is integer 10
		It is float 10.1
		It is string Ten

34. ஆக்கி பணிமிகுப்பு (Constructor overloading)

```
#include<iostream>
using namespace std;
class add
{
 int num1, num2, sum;
public:
add()
{
cout<<"\n Constructor without parameters..
";
num1= 0; num2= 0; sum = 0;
 }
add ( int s1, int s2 )
{
 cout<<"\n Parameterized constructor... ";
 num1= s1; num2=s2; sum=0;
}
add (add &a)
{
cout<<"\n Copy Constructor ... ";
 num1= a.num1;
 num2=a.num2;
 sum = 0;
}
void getdata()
{
cout<<"\nEnter data ... ";
cin>>num1>>num2;
}
void addition()
{
 sum=num1+num2;
}
void putdata()
 {
cout<<"\n The numbers are..";
cout<<num1<<'\t'<<num2;
cout<<"\n The sum of the numbers are.. "<<
sum;
}
};
int main()
 {
add a, b (10, 20) , c(b);
a.getdata();
a.addition();
b.addition();
c.addition();
cout<<"\n Object a : ";
a.putdata();
cout<<"\n Object b : ";
b.putdata();
cout<<"\n Object c.. ";
c.putdata();
return 0;
}
```

```
Constructor without
parameters..
 Parameterized constructor...
 Copy Constructor ...
Enter data ... 15
20

 Object a :
 The numbers are..15    20
 The sum of the numbers are..
35
 Object b :
 The numbers are..10    20
 The sum of the numbers are..
30
 Object c..
 The numbers are..10    20
 The sum of the numbers are..
30
```

35. செயற்குறி பணிமிகுப்பு (operator overloading)

```
#include<iostream>
using namespace std;
class complex
{
 int real,img;
 public:
void read()
{
cout<<"\nEnter the REAL PART : ";
cin>>real;
cout<<"\nEnter the IMAGINARY PART : ";
cin>>img;
}
complex operator +(complex c2)
{
complex c3;
c3.real=real+c2.real;
c3.img=img+c2.img;
return c3;
}
void display()
{
cout<<real<<"+"<<img<<"i";
}
};
int main()
{
complex c1,c2,c3;
int choice, cont;
cout<<"\n\nEnter the First Complex
Number";
c1.read();
cout<<"\n\nEnter the Second Complex
Number";
c2.read();
c3=c1+c2; // binary + overloaded
cout<<"\n\nSUM = ";
c3.display();
return 0;
 }
```

```
Enter the First Complex
Number
Enter the REAL PART : 3

Enter the IMAGINARY PART :
4


Enter the Second Complex
Number
Enter the REAL PART : 5

Enter the IMAGINARY PART :
8


SUM = 8+12i
```

36. எடுத்துக்காட்டு: ஒரு வழி மரபுரிமம் (Single inheritance)

```
# include <iostream>
using namespace std;
class student //base class
{
private :
char name[20];
int rno;
public:
void acceptname()
{
cout<<"\n Enter roll no and name .. ";
cin>>rno>>name;
}
void displayname()
{
cout<<"\n Roll no :-"<<rno;
cout<<"\n Name :-"<<name<<endl;
}
```

```
Enter roll no and name .. 12101
KUMAR

 Enter lang,eng,phy,che,csc,mat
marks.. 99 97 52 65 89 95 90

 Roll no :-12101
 Name :-KUMAR

             Marks Obtained
 Language.. 99
 English .. 97
 Physics .. 52
 Chemistry.. 65
 Comp.sci.. 89
 Maths .. 95
```

	};
	class exam : public student //derived class with single base class
	{
	public:
	int mark1, mark2 ,mark3,mark4,mark5,mark6,total;
	void acceptmark()
	{
	cout<<"\n Enter lang,eng,phy,che,csc,mat marks.. ";
	cin>>mark1>>mark2>>mark3>>mark4>>mark5>>mark6;
	}
	void displaymark()
	{
	cout<<"\n\t\t Marks Obtained ";
	cout<<"\n Language.. "<<mark1;
	cout<<"\n English .. "<<mark2;
	cout<<"\n Physics .. "<<mark3;
	cout<<"\n Chemistry.. "<<mark4;
	cout<<"\n Comp.sci.. "<<mark5;
	cout<<"\n Maths .. "<<mark6;
	}
	};
	int main()
	{
	exam e1;
	e1.acceptname(); //calling base class function using derived class object
	e1.acceptmark();
	e1.displayname(); //calling base class function using derived class object
	e1.displaymark();
	return 0;
	}	

37. ஆக்கிகள், அழிப்பிகள் வரிசைமுறை (The order of constructor & Destructor)	#include<iostream>
	using namespace std;
	class base
	{
	public:
	base()
	{
	cout<<"\n Constructor of base class...";
	}
	~base()
	{
	cout<<"\n Destructor of base class.... ";
	}
	};
	class derived:public base
	{
	public :
	derived()
	{
	cout << "\n Constructor of derived ...";
	}
	~derived()
	{
	cout << "\n Destructor of derived ...";
	}
		Constructor of base class...
		Constructor of derived ...
		Constructor of derived1 ...
		Destructor of derived2 ...
		Destructor of derived ...
		Destructor of base class....

	};
	class derived1 :public derived
	{
	public :
	derived1()
	{
	cout << "\n Constructor of derived1 ...";
	}
	~derived1()
	{
	cout << "\n Destructor of derived2 ...";
	}
	};
	int main()
	{
	derived1 x;
	return 0;
	}	

38.பின்வரும் எண் தொடரை அச்சிடுவதற்கான நிரல்களை எழுதுக.
1 3 5 7...... 75
June 2019
	#include<iostream>
	using namespace std;
	int main( )
	{
	for (int i=1; i<=75 ; i+=2)
	cout << i<<' ' ;
	return 0;
	}
		1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59 61 63 65 67 69 71 73 75

39. பின்வரும் எண் தொடரை அச்சிடுவதற்கான நிரல்களை எழுதுக.
(While loop)
5,10,15,20,.....50,
Mar - 2019
	#include<iostream>
	using namespace std;
	int main()
	{
	int i=5;
	while(i<=50)
	{
	cout <<i<<" ," ;
	i=i+5;
	}
	return 0;
	}
		5 ,10 ,15 ,20 ,25 ,30 ,35 ,40 ,45 ,50 ,

40. பின்வரும் எண் தொடரை அச்சிடுவதற்கான நிரல்களை எழுதுக
(for loop)
21 22 23 24 25......30
sep -2021
	#include<iostream>
	using namespace std;
	int main()
	{
	int i;
	for(i=21;i<=30;i++)
	cout <<i<<" " ;
	return 0;
	}
		21 22 23 24 25 26 27 28 29 30

Prepared by S. Saminathan M.C.A.,B.Ed.,M.Phil.,
Computer Instructor Grade I
GHSS – MUKHASAPARUR
CUDDALORE DT.
ss79nathan@gmail.com