**Term work for MID-TERM**

**on**

**[ OOPs with C++ ]**

**(PCS 307)**

**2021-22**

**Submitted to: Submitted by:**

Mr. Akash Chauhan Nishant Bhandari

-Assistant Professor University Roll. No.: 2018525 GEHU, D.Dun Class Roll. No./Section: 35/C

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GRAPHIC ERA HILL UNIVERSITY, DEHRADUN**

**ACKNOWLEDGMENT**

I would like to particularly thank my Data Science using OOPs with C++ Lab Faculty Mr. Akash Chauhan for his patience, support and encouragement throughout the completion of this Term work.

At last but not the least I greatly indebted to all other persons who directly or indirectly helped me during this course.

Photo [optional]  **Nishant Bhandari**

**University. Roll No.-** 2018525

**B.Tech CSE-C-III Sem**

**Session: 2021-22**

**GEHU, Dehradun**



Table of Contents

|  |  |  |
| --- | --- | --- |
| **Program No.** | **Program Name** | **Page No** |
| 1 | Run all four compilation units individually for any sample program using C++. |  |
| 2 | Write an Efficient code to check if a number is prime or not. |  |
| 3 | Write C++ code for below mentioned tasks?  **Task1:** How the preprocessor will react when you try to use, #include<iostream.h> instead of #include<iostream>?  **Task2:** How the preprocessor will react when you use cout but don't include #include<iostream> in your code?  **Task3:** Take a char variable and use cin to take its value from the user, cout it and it will only return one character and loss rest of the data you have entered!  **Task4:** How to resolve above issue? [Hint: by using, getline(cin, line);] |  |
| 4 | Write C++ code for below mentioned tasks?  **add(int,int), add(float, float), both of these methods are in two different namespaces First and Second respectively.**  **Task1:** Access these methods using scope resolution operator [::](SRO) from main method?  **Task2:** Access these methods using "using" keyword for main method?  **Task3:**  Try to access these methods without using, (SRO) and "using" keyword and check how the compiler will react to it?  **Task4:** Try to access these methods for Mixed Values [Int, Float] and see how the compiler will react to it? |  |
| 5 | Write C++ code for below mentioned tasks?  **Primary Data Types related questions in C++:**  **Task1:**Initialise all primary data types, assign their values and print them all? [char, bool, short, int, long, float, double, long double, wide char]  **Task2:** Apply sizeof operator on all above operators and their variables?        Example        Int a = 10;       Cout << Sizeof(int);       Cout << Sizeof(a); |  |
| 6 | Write C++ code for below mentioned tasks?  **String related Questions in C++:**  **Task1:**What happens if we add integer with a string, how the compiler would react to it?  String str = "ABC"; Int a = 1; String str2 = str + a;  **Task2:** Check the entered string is Palindrome or not?  String str = "75457" Output: Yes it is a palindrome or No it is not a Palindrome.  (Use, getline(sin, str1) and reverse\_iterator of string to check this)  **Task3:** Make a reverse of a string using reverse method and reverse\_iterator of string class?  **Task4:** String Compare: Check if the strings are equal or not? (do not use str1.compare(str2), do it manually)  **Task5:** String Compare: Check the possible values string.compare() function will return?              (Create cases in which compare function would return below values)  X>0 X<0 X==0 X = -4 X = 5 X = -2104040...  Also check the ASCII difference between two characters?(use int type cast)  **Task6:** Check if string is mutable in C++ or not? String a = "Hello"; Cout << &a; a[0] = 'J'; Cout << &a; Cout << a;  What is the output? |  |
| 7 | Write C++ code for below mentioned tasks?  **Array and 2D Array related Questions in C++:**  **Task1:** Create a switch statement [Manual], In Which:    a. When you pass 1 your program would print current year   b. When you pass 2 your program would print current month   c. When you pass 3 your program would print current day   d. When you pass 4 your program would print Not applicable  **Task2:** Create a switch statement [Using ctime], In Which:    a. When you pass 1 your program would print current year   b. When you pass 2 your program would print current month   c. When you pass 3 your program would print current day   d. When you pass 4 your program would print Not applicable    **Task3:**    v1. Print using reverse method:    1 2 3  9 8 7    4 5 6  ==>    6 5 4    7 8 9             3 2 1      v2. Print using (10- arr[i][j])  method:    1 2 3  9 8 7    4 5 6  ==>    6 5 4    7 8 9             3 2 1      v3. Restore using reverse method [without creating new array]:    1 2 3  9 8 7    4 5 6  ==>    6 5 4    7 8 9             3 2 1      v4. Restore using (10- arr[i][j]) method [without creating new array]:    1 2 3  9 8 7    4 5 6  ==>    6 5 4    7 8 9              3 2 1  **Task4:** Restore the same values in the same array, arr[3][3]:    1 2 3 1 1 1            4 5 6     ==> 2 2 2    7 8 9 3 3 3    v1. Use row loop [int i, for all j]    v2. Use arr[i][N-1]/3, at each place    v3. Use, arr[i][j]-(2\*i+j)  **Task5:** Store these in an array[4][4] in given fassion and then print:  \*  \* \*  \* \* \*  \* \* \* \*  **Task6:** Store these in an array[4][4] in given fashion and then print:  \* \* \* \*  \* \* \*  \* \*  \*  **Task7:** Store these in an array[4][4] in given fashion and then print:  \*          \* \*        \* \* \*      \* \* \* \*    **Task8:** Store these in an array[4][4] in given fashion and then print:  \* \* \* \*  \* \* \*  \* \*  \* |  |
| 8 | Write C++ code for below mentioned tasks?  **Pointer, Function, Inline Function, Recursion in C++:**  **Task1:**Will the program through an error and if yes then why?  int \*p = {10,20,20};  cout << \*p;  p++;  cout << \*p;  **Task2:**Output of this program?  **V1.** Issue?  int arr[] = {10,20,30};  cout << \*arr;  cout << arr;  arr++;  cout << \*arr;  **V2.**How to resolve above issue?  int arr[] = {10,20,30};  cout << \*arr;  cout << arr;  cout << \*(?);  **Task3:**Output of this program?  **V1.**Output?  int a = 10;  int \*p;  int \*\*q;  p = &a;  q = &p;  cou << \*p;  cou << \*\*q;  **V2.** Change the value of a using q pointer to pointer.  **Task4:** Find factorial of a number using function but not recurssion  **Task5:** Find factorial of a number using recurssion  **Task6:** Series Problem using recurssion for n series  2, (2^2 + 2), (3^3 + 3), (4^4 + 4), (5^5 + 5), .......  Hint:  n \* ((n-1)^(n-1) + (n-1))  **Task7:** Perform Call by value, call by Address for swapping value of a and b:  int a = 10;  int b = 20;  **V1.**Swap(a,b); //call by Value [void swap(int a, int b){}]  **V2.**Swap(a,b); //call by Value [void swap(int &a, int &b){}]  **V3.**Swap(&a,&b); //call by Address |  |
| 9 | Write C++ code for below mentioned tasks?  **Class, Object, Constructor, Static Data Members, friend function in C++:**  **Task1: Class and Object in C++** a. WAP to assign and print the roll number, phone number and address of two students having names "Sam" and "John" respectively by creating two objects of the class 'Student'. b. WAP which would contain array of objects [many objects], of a class Student. Student [Name, Age, Year, section, marks], the section would be A,B,C and D. Your program would be able to return the total marks of students in the college. Hint [Make a Matrix or Tabular diagram to understand the problem], all the rows will differ each other by different objects of Student class [Student s1,s2,s3,s4].  **Task2: Constructor in C++** WAP to create a class to print the area of a square and a rectangle. The class has two functions with the same name but different number of parameters. The function for printing the area of rectangle has two parameters which are its length and breadth respectively while the other function for printing the area of square has one parameter which is the side of the square. Use multiple constructors to for the initialization.  **Task3: Static Data Members in C++** WAP to count the total number of calls for a member function from more than one objects. [Lets say, from 3 such Objects]  **Task4: Friend Function in C++** WAP in which you create a Student class having basic information for each student, like name, age and marks. By using friend function add marks of all the students [lets say 3 objects] and print it.  **Task5: Structure in C++** WAP to create a College class and Student Structure in C++ in one program. By providing such suitable examples write at least 5 differences between class and struct code your have written above.  Hint [Access Specifiers, Heap and Stack, large and small memory, etc.]  **Task6: Extra Questions:** WAP which would perform these tasks of your data: a. Come to next line b. set minimum field width c. fill string with (\*) after setw(15) function     \*\*\*\*\*\*\*\*\*\*1234 by using endl, setw, and setfill [Manipulators in C++] |  |
| 10 | Write C++ code for below mentioned tasks?  **Array of Objects, Pointer to Object, This pointer, Operator Overloading in C++**  **Task1: Array of Objects in C++** WAP to create a directory that contains the following information. (a) Name of a person (b) Address (c) Telephone Number (if available with STD code) (d) Mobile Number (if available) (e) Head of the family  **Task2: Pointer to Object in C++** WAP to create print or display Student information containing in Student class by using pointers to object.  **Task3: This pointer in C++** WAP to pass two variables in a parameterized constructor during object creation and have same names variables as class member data and constructor parameters. Your job is to calculate the remainder of those two numbers.  **Task4: Operator Overloading in C++** a). WAP, in which you write a friend function to overload a less than '<' operator in C++. b). WAP in which you can add two objects [every object would have 1 integer value] by overloading + operator, which eventually would add the data values of those two object by adding the objects. |  |



**DEPARTMENT OF CSE**

|  |
| --- |
| Photograph  Passport Size |

**B.Tech. CSE**

**STUDENT LAB REPORT SHEET**

**Name of Student ……………………………………….…….Mob.No……………………………….…………..**

**Address Permanent ………………………………………………………………………………………….………..**

**Father’s Name …………………………Occupation ……………………………MoNo……..…….….………**

**Mother’s Name ………………… …..Occupation……………………………MoNo…………….…………..**

**Section …………….Branch……………………Semester……………..Class Roll No………………. Grade A B C**

**Local Address…………………………………………………Email……………….……………………. Marks 5 3 1**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S.No.** | **Practical** | **D.O.P.** | **Date of Submission** | **Grade (Viva)** | **Grade (Report File)** | **Total Marks (out of 10)** | **Student’s Signature** | **Teacher’s Signature** |
| **1** | **Practical-01** |  |  |  |  |  |  |  |
| **2** | **Practical-02** |  |  |  |  |  |  |  |
| **3** | **Practical-03** |  |  |  |  |  |  |  |
| **4** | **Practical-04** |  |  |  |  |  |  |  |
| **5** | **Practical-05** |  |  |  |  |  |  |  |
| **6** | **Practical-06** |  |  |  |  |  |  |  |
| **7** | **Practical-07** |  |  |  |  |  |  |  |
| **8** | **Practical-08** |  |  |  |  |  |  |  |
| **9** | **Practical-09** |  |  |  |  |  |  |  |
| **10** | **Practical-10** |  |  |  |  |  |  |  |
| **11** |  |  |  |  |  |  |  |  |
| **12** |  |  |  |  |  |  |  |  |

**Program 1**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**int main( ){**

**cout << "Running" << endl ;**

**return 0;**

**}**

**Program 1**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 2**

**Source Code:**

**#include <iostream>**

**#include <cmath>**

**using namespace std ;**

**int chk( int n ){**

**if( n == 2 ) return 1 ;**

**if( n == 1 || n % 2 == 0 ) return 0 ;**

**// Now check for Odd nums. only**

**for( int i = 3; i <= sqrt( n ); i += 2 ){**

**if( n % i == 0 )**

**return 0 ;**

**}**

**return 1 ;**

**}**

**int main(){**

**int n ;**

**cin >> n ;**

**// Check for Prime**

**if( chk( n ) )**

**cout << n << " is Prime" << endl ;**

**else**

**cout << n << " is not Prime" << endl ;**

**return 0 ;**

**}**

**Program 2**

**A screenshot of a computer

Description automatically generatedOutput:**

**Program 3.1**

**Source Code:**

**#include <iostream.h>**

**using namespace std ;**

**int main(){**

**cout << "Running" << endl ;**

**return 0;**

**}**

**Program 3.1**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 3.2**

**Source Code:**

**// #include <iostream>**

**using namespace std ;**

**int main(){**

**cout << "Running" << endl ;**

**return 0;**

**}**

**Program 3.2**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 3.3**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**int main(){**

**char c ;**

**cin >> c ;**

**cout << "c: " << c << endl ;**

**string str ;**

**getline( cin, str ) ;**

**cout << "str: " << str << endl ;**

**return 0;**

**}**

**Program 3.3**

**Output:Graphical user interface, text

Description automatically generated**

**Graphical user interface, text

Description automatically generated**

**Program 4.1**

**Source Code:**

**#include<iostream>**

**namespace first{**

**int add( int x, int y ){**

**return x + y ;**

**}**

**}**

**namespace second{**

**float add( float x, float y ){**

**return x + y ;**

**}**

**}**

**int main(){**

**std :: cout << first :: add( 2,7 ) << std :: endl ;**

**std :: cout << second :: add( 2.2,7.9 ) << std :: endl ;**

**return 0;**

**}**

**Program 4.1**

**Output:**

**Text

Description automatically generated**

**Program 4.2**

**Source Code:**

#include<iostream>

using namespace std ;

namespace first{

int add( int x, int y ){

cout << "In Int " ;

return x + y ;

}

}

namespace second{

float add( float x, float y ){

cout << "In Float " ;

return x + y ;

}

}

using namespace first ;

using namespace second ;

int main(){

cout << add( 2, 4 ) << endl ;

cout << add( 2, 4.2 ) << endl ;

cout << add( 2.2, 4 ) << endl ;

cout << add( 2.2f, 4.2f ) << endl ; // By Default float is double

return 0;

}

**Program 4.2**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 4.3**

**Source Code:**

#include<iostream>

using namespace std ;

namespace first{

int add( int x, int y ){

cout << "In Int " ;

return x + y ;

}

}

namespace second{

float add( float x, float y ){

cout << "In Float " ;

return x + y ;

}

}

int main(){

cout << add( 2, 4 ) << endl ;

cout << add( 2, 4.2 ) << endl ;

cout << add( 2.2, 4 ) << endl ;

cout << add( 2.2f, 4.2f ) << endl ;

return 0;

}

**Program 4.3**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 5.1**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**void char\_func( void ){**

**char c = 'A' ;**

**cout << "Char: " << c << endl ;**

**}**

**void bool\_func( void ){**

**bool val = ( 10 > 9 ) ;**

**cout << "Bool: " << val << endl ;**

**}**

**void short\_func( void ){**

**short sh = 10 ;**

**cout << "Short: " << sh << endl ;**

**}**

**void int\_func( void ){**

**int in = 643528 ;**

**cout << "Integer: " << in << endl ;**

**}**

**void long\_func( void ){**

**long lng = 3475928357 ;**

**cout << "Long: " << lng << endl ;**

**}**

**void float\_func( void ){**

**float fl = 34.7592835f ;**

**cout << "Float: " << fl << endl ;**

**}**

**void double\_func( void ){**

**double db = 34.7592835789 ;**

**cout << "Double: " << db << endl ;**

**}**

**void longdouble\_func( void ){**

**long double ldb = 34.75928357657475689L ;**

**cout << "Long Double: " << ldb << endl ;**

**}**

**void widechar\_func( void ){**

**wchar\_t w = L'ग' ;**

**cout << "Wide Char: " << w << endl ;**

**// Use wcout to print the original char**

**}**

**int main(){**

**char\_func( ) ;**

**bool\_func( ) ;**

**short\_func( ) ;**

**int\_func( ) ;**

**long\_func( ) ;**

**float\_func( ) ;**

**double\_func( ) ;**

**longdouble\_func( ) ;**

**widechar\_func( ) ;**

**return 0;**

**}**

**Program 5.1**

**Output:**

Text

Description automatically generated

**Program 5.2**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**void char\_func( void ){**

**char ch = 'A' ;**

**cout << "Char: " << ch << endl ;**

**cout**

**<< "size of var : " << sizeof(ch) << "// size of datatype : "**

**<< sizeof(char) << "// size of value : " << sizeof('A') << "\n"**

**<< endl;**

**}**

**void bool\_func( void ){**

**bool val = ( 10 > 9 ) ;**

**cout << "Bool: " << val << endl ;**

**cout << "size of var : " << sizeof( val ) << "// size of datatype : "**

**<< sizeof(bool) << "// size of value : " << sizeof('1') << "\n"**

**<< endl;**

**}**

**void short\_func( void ){**

**short sh = 10 ;**

**cout << "Short: " << sh << endl ;**

**cout**

**<< "size of var : " << sizeof(sh) << "// size of datatype : "**

**<< sizeof(short) << "// size of value : " << sizeof( 10 ) << "\n"**

**<< endl ;**

**}**

**void int\_func( void ){**

**int in = 643528 ;**

**cout << "Integer: " << in ;**

**cout << " // size of value : " << sizeof( in ) << endl;**

**}**

**void long\_func( void ){**

**long lng = 34759283 ;**

**cout << "Long: " << lng ;**

**cout << " // size of value : " << sizeof( lng ) << endl;**

**}**

**void float\_func( void ){**

**float fl = 34.7592835f ;**

**cout << "Float: " << fl ;**

**cout << " // size of value : " << sizeof( fl ) << endl;**

**}**

**void double\_func( void ){**

**double db = 34.7592835789 ;**

**cout << "Double: " << db ;**

**cout << " // size of value : " << sizeof( db ) << endl;**

**}**

**void longdouble\_func( void ){**

**long double ldb = 34.75928357657475689L ;**

**cout << "Long Double: " << ldb ;**

**cout << " // size of value : " << sizeof( ldb ) << endl;**

**}**

**void widechar\_func( void ){**

**wchar\_t w = L'ग' ;**

**cout << "Wide Char: " << w ;**

**cout << " // size of value : " << sizeof( w ) << endl;**

**// Use wcout to print the original char**

**}**

**int main(){**

**char\_func( ) ;**

**bool\_func( ) ;**

**short\_func( ) ;**

**int\_func( ) ;**

**long\_func( ) ;**

**float\_func( ) ;**

**double\_func( ) ;**

**longdouble\_func( ) ;**

**widechar\_func( ) ;**

**return 0;**

**}**

**Program 5.2**

**Output:**

Text

Description automatically generated

**Program 6.1**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**int main(){**

**int a = 10, b = 5 ;**

**// cout << a << " + " << b << " = " << a + b << endl ; 15**

**// cout << "str" + "ing" << endl ;**

**cout << a + "string" << endl ;**

**return 0 ;**

**}**

**z**

**Program 6.1**

**Output:**

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

**Program 6.2**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**bool xrev( string s ){**

**int len = s.length();**

**for( int i = 0; i < len/2; i++ ){**

**if( s[ i ] != s[ len -1 - i ] )**

**return false ;**

**}**

**return true ;**

**}**

**int main(){**

**string str ;**

**getline( cin, str ) ;**

**bool ans = xrev( str ) ;**

**cout << ans << endl ;**

**return 0 ;**

**}**

**Program 6.2**

**Output:**

Text

Description automatically generated

**Program 6.3**

**Source Code:**

**#include <iostream>**

**#include <string>**

**using namespace std ;**

**int main(){**

**string s ;**

**getline( cin, s ) ;**

**string rev ;**

**string :: iterator i ;**

**for( i = s.end(); i >= s.begin() ; i-- ){**

**rev.push\_back( \*i ) ;**

**}**

**cout << "Reverse is: " ;**

**cout << rev << endl << endl ;**

**return 0 ;**

**}**

**Program 6.3**

**Output:**

Graphical user interface, text

Description automatically generated

**Program 6.4**

**Source Code:**

**#include <iostream>**

**#include <string>**

**using namespace std ;**

**int main(){**

**string s1;**

**cout << "Enter String 1: " ;**

**getline( cin, s1 ) ;**

**string s2;**

**cout << "Enter String 2: " ;**

**getline( cin, s2 ) ;**

**// cout << s1.length() << "|" << s2.length() << endl ;**

**if( s1.length() != s2.length() ) { // base Case**

**cout << "Not Equal\n" << endl ;**

**return 0 ;**

**}**

**string :: iterator i1 = s1.begin() ;**

**string :: iterator i2 = s2.begin() ;**

**while( i1 != s1.end() ){**

**if( \*i1 != \*i2 )**

**break ;**

**i1++ ;**

**i2++ ;**

**}**

**if( i1 != s1.end() )**

**cout << "Not Equal\n" << endl ;**

**else**

**cout << "Equal\n" << endl ;**

**return 0 ;**

**}**

**Program 6.4**

**Output:**

A screenshot of a computer

Description automatically generated

**Program 6.5**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**int main()**

**{**

**string s1 ;**

**string s2 ;**

**cin >> s1 >> s2 ;**

**int x=s1.compare(s2);**

**if( x == 0 )**

**cout << "EQUAL" << endl;**

**else if( x > 0 )**

**cout << "LARGER " << endl;**

**else**

**cout << "SMALLER" << endl;**

**return 0;**

**}**

**Program 6.5**

**Output:**

Text

Description automatically generated

**Program 6.6**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**int main(){**

**string s = "Hello" ;**

**s[0] = 'J';**

**cout << s << endl ;**

**return 0;**

**}**

**Program 6.6**

**Output:**

Graphical user interface, text, application

Description automatically generated

**Program 7.1**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int main(){**

**int ch ;**

**cout<<"Enter\n1. To check current day\n2. To check current month\n3. To check current month\n";**

**cout<<"Enter your choice: ";**

**cin>>ch;**

**switch(ch)**

**{**

**case 1: cout<<"The current Day (Manually) is: "<<"MON"<<endl;**

**break;**

**case 2: cout<<"The Current Month(Manually) is: "<<"OCT"<<endl;**

**break;**

**case 3: cout<<"The Current Year(Manually) is: "<<2021<<endl;**

**break;**

**default: cout<<"Invalid Choice\n";**

**}**

**return 0;**

**}**

**Program 7.1**

**Output:**

Text

Description automatically generated

**Program 7.2**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int main(){**

**int f, ch ,l, m;**

**time\_t curr;**

**curr=time(NULL);**

**string c=ctime(&curr);**

**f=c.find\_first\_of(" ");**

**cout<<"Enter\n1. To check current day\n2. To check current month\n3. To check current Year\n";**

**cout<<"Enter your choice: ";**

**cin>>ch;**

**switch(ch){**

**case 1: cout << "The current Day (using ctime) is: " << c.substr(0,f) << endl;**

**break;**

**case 2: m = (c.substr(f+1)).find\_first\_of(" ");**

**cout<<"The Current Month(Using Ctime) is:"<< c.substr(f,m+1) <<endl;**

**break;**

**case 3: l = c.find\_last\_of(" ");**

**cout<<"The current year(Using ctime) is: "<< c.substr(l+1) <<endl;**

**break;**

**default: cout<<"Invalid Choice\n";**

**}**

**cout << endl ;**

**return 0 ;**

**}**

**Program 7.2**

**Output:**

Text

Description automatically generated

**Program 7.3**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**int main(){**

**int arr[ 3 ][ 3 ] = { { 1, 2, 3 },**

**{ 4, 5, 6 },**

**{ 7, 8, 9 }**

**} ;**

**cout << "Before: " << endl ;**

**for( int i = 0; i < 3; i++ ){**

**for( int j = 0; j < 3; j++ ){**

**cout << arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**cout << "After: " << endl ;**

**for( int i = 2; i >= 0; i-- ){**

**for( int j = 2; j >= 0; j-- ){**

**cout << arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**cout << endl ;**

**return 0 ;**

**}**

**Program 7.3**

**Output:**

Graphical user interface

Description automatically generated

**Program 7.4**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**int main(){**

**int arr[][ 3 ] = { { 1, 2, 3 },**

**{ 4, 5, 6 },**

**{ 7, 8, 9 }**

**} ;**

**cout << "Before: " << endl ;**

**for( int i = 0; i < 3; i++ ){**

**for( int j = 0; j < 3; j++ ){**

**cout << arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**cout << "After: " << endl ;**

**for( int i = 0; i < 3; i++ ){**

**for( int j = 0; j < 3; j++ ){**

**cout << 10 - arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**cout << endl ;**

**return 0 ;**

**}**

**Program 7.4**

**Output:**

A picture containing text

Description automatically generated

**Program 7.5**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**void swap( int \*p, int \*q ){**

**int t = \*p ;**

**\*p = \*q ;**

**\*q = t ;**

**}**

**int main(){**

**int n = 3 ;**

**int arr[ n ][ n ] = { { 1, 2, 3 },**

**{ 4, 5, 6 },**

**{ 7, 8, 9 }**

**} ;**

**cout << "Before: " << endl ;**

**for( int i = 0; i < n; i++ ){**

**for( int j = 0; j < n; j++ ){**

**cout << arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**// SWAP Cols**

**for( int i = 0; i < n; i++ ){**

**for( int j = 0; j <= n /2; j++ ){**

**swap( &arr[ i ][ j ], &arr[ i ][ n -1 -j ] ) ;**

**}**

**}**

**// 3 2 1 => 9 8 7**

**// 6 5 4 => 6 5 4**

**// 9 8 7 => 3 2 1**

**// SWAP Rows**

**for( int i = 0; i < n /2; i++ ){**

**for( int j = 0; j < n; j++ ){**

**swap( &arr[ i ][ j ], &arr[ n -1 -i ][ j ] ) ;**

**}**

**}**

**cout << "After: " << endl ;**

**for( int i = 0; i < n; i++ ){**

**for( int j = 0; j < n; j++ ){**

**cout << arr[ i ][ j ] << " " ;**

**}**

**cout << endl ;**

**}**

**cout << endl ;**

**return 0 ;**

**}**

**Program 7.5**

**Output:**

Graphical user interface

Description automatically generated with medium confidence

**Program 7.6**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**int main(){**

**int n ;**

**cin >> n ;**

**for( int i = 1; i <= n; i++ ){**

**for( int j = 1; j <= i; j++ ){**

**cout << "\* " ;**

**}**

**cout << endl ;**

**}**

**return 0 ;**

**}**

**Program 7.6**

**Output:**

Text

Description automatically generated

**Program 7.7**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**int main(){**

**int n ;**

**cin >> n ;**

**for( int i = n-1; i >= 0; i-- ){**

**int j ;**

**for( j = 1; j <= i; j++ ){**

**cout << " " ;**

**}**

**for( ; j <= n; j++ ){**

**cout << "\* " ;**

**}**

**cout << endl ;**

**}**

**return 0 ;**

**}**

**Program 7.7**

**Output:**

A picture containing text

Description automatically generated

**Program 7.8**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**int main(){**

**int n ;**

**cin >> n ;**

**for( int i = n; i > 0; i-- ){**

**int j ;**

**for( j = n; j > i; j-- ){**

**cout << " " ;**

**}**

**for( ; j > 0; j-- ){**

**cout << "\* " ;**

**}**

**cout << endl ;**

**}**

**return 0 ;**

**}**

**Program 7.8**

**Output:**

Text

Description automatically generated

**Program 7.9**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**int main(){**

**int n ;**

**cin >> n ;**

**for( int i = n; i > 0; i-- ){**

**int j ;**

**for( j = 0; j < i; j++ ){**

**cout << "\* " ;**

**}**

**for( ; j <= n; j++ ){**

**cout << " " ;**

**}**

**cout << endl ;**

**}**

**return 0 ;**

**}**

**Program 7.9**

**Output:**

Text

Description automatically generated

**Program 8.1**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**int main(){**

**int \*p={10,20,30};**

**cout<<\*p<<" " ;**

**p++;**

**cout<<\*p;**

**return 0;**

**}**

**Program 8.1**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 8.2**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int main(){**

**int p[] = { 10, 20, 30 } ;**

**cout << \*p << endl ;**

**// Output 1 when p is not incremented**

**// p++ ;**

**// cout << \*p << endl ;**

**// Above issue can be resolved using the below syntax**

**cout << \*( p +1 ) << endl ;**

**return 0 ;**

**}**

**// HEre p is a Static pointer**

**// i.e p can point to a memory only once after that no reassignment is possible**

**// That's why compiler is throwing lvalue error**

**Program 8.2**

**Output:**

Text

Description automatically generated

**Program 8.3**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int main(){**

**int a = 20 ;**

**// V1.**

**int \*p ;**

**p = &a ;**

**int \*\*q = &p ;**

**cout << \*p << endl ;**

**cout << \*\*q << endl ;**

**// V2.**

**\*\*q = 10 ;**

**cout << a << endl ;**

**return 0 ;**

**}**

**Program 8.3**

**Output:**

Graphical user interface

Description automatically generated with medium confidence

**Program 8.4**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int fact( int n ) {**

**int ans = 1 ;**

**for( int i = 2; i <= n; i++ ){**

**ans \*= i ;**

**}**

**return ans ;**

**}**

**int main(){**

**int ans ;**

**ans = fact( 5 ) ;**

**cout << "Fact: " << ans << endl ;**

**return 0 ;**

**}**

**Program 8.4**

**Output:**

Graphical user interface

Description automatically generated with medium confidence

**Program 8.5**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**int fact( int n ) {**

**if( n <= 1 ){**

**return 1 ;**

**}**

**return n \* fact( n -1 ) ;**

**}**

**int main(){**

**int ans , n ;**

**cout << "Enter a NUmber: " ;**

**cin >> n ;**

**ans = fact( n ) ;**

**cout << "\nFact of " << n << " is: " << ans << endl ;**

**return 0 ;**

**}**

**Program 8.5**

**Output:**

Text

Description automatically generated

**Program 8.6**

**Source Code:**

**#include <iostream>**

**using namespace std;**

**void seq ( int n ) {**

**if ( n == 1 ) {**

**return ;**

**}**

**seq ( n -1 ) ;**

**cout << n\* ( pow(n-1,n-1) +n -1 ) << " " ;**

**return ;**

**}**

**int main(){**

**int n ;**

**cin >> n ;**

**seq( n ) ;**

**return 0 ;**

**}**

**Program 8.6**

**Output:**

Text

Description automatically generated

**Program 8.7a**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**void swap(int a,int b)**

**{**

**int t=a;**

**a=b;**

**b=t;**

**}**

**int main(){**

**int a=10,b=20;**

**cout<<"Values before swap: "<<a<<"\t"<<b<<endl;**

**swap(a,b);**

**cout<<"Values after swap: "<<a<<"\t"<<b<<endl;**

**return 0;**

**}**

**Program 8.7a**

**Output:**

Text

Description automatically generated

**Program 8.7b**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**void swap(int &a,int &b)**

**{**

**int t=a;**

**a=b;**

**b=t;**

**}**

**int main(){**

**int a=10,b=20;**

**cout<<"Values before swap: "<<a<<"\t"<<b<<endl;**

**swap( a,b );**

**cout<<"Values after swap: "<<a<<"\t"<<b<<endl;**

**return 0;**

**}**

**Program 8.7b**

**Output:**

Text

Description automatically generated

**Program 8.7c**

**Source Code:**

**#include<iostream>**

**using namespace std ;**

**void swap(int \*a,int \*b)**

**{**

**int t = \*a ;**

**\*a = \*b ;**

**\*b = t ;**

**}**

**int main(){**

**int a=10,b=20;**

**cout<<"Values before swap: "<<a<<"\t"<<b<<endl;**

**swap( &a, &b );**

**cout<<"Values after swap: "<<a<<"\t"<<b<<endl;**

**return 0;**

**}**

**Program 8.7c**

**Output:**

Text

Description automatically generated with medium confidence

**Program 9.1a**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class Student {**

**int roll ;**

**unsigned long long num ;**

**string addr ;**

**public:**

**Student( int roll, string addr, unsigned long long num ){**

**this->roll = roll ;**

**this->num = num ;**

**this->addr = addr ;**

**}**

**void show( void ){**

**cout << "Roll no.: " << roll << endl ;**

**cout << "Mobile no.: " << num << endl ;**

**cout << "Address.: " << addr << endl ;**

**}**

**};**

**int main( ){**

**Student Sam ( 45, "Clement Town", (unsigned long long) 7868754329 ) ;**

**Student John ( 23, "Shubhash Nagar", (unsigned long long) 9842332145 ) ;**

**Sam.show() ;**

**cout << endl ;**

**John.show() ;**

**return 0;**

**}**

**Program 9.1a**

**Output:**

A screenshot of a computer

Description automatically generated

**Program 9.1b**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class Student {**

**string name ;**

**short age ;**

**short year ;**

**char sec ;**

**float marks ;**

**public:**

**void assign( string name, short age, short year, char sec, float marks ){**

**this-> name = name ;**

**this-> age = age ;**

**this-> year = year ;**

**this-> sec = sec ;**

**this-> marks = marks ;**

**}**

**void show( void ){**

**cout << "Name.: " << name << endl ;**

**cout << "Age.: " << age << endl ;**

**cout << "Year.: " << year << endl ;**

**cout << "Section.: " << sec << endl ;**

**cout << "Marks.: " << marks << endl ;**

**cout << endl ;**

**}**

**friend float showM( Student \* ) ;**

**};**

**float showM( Student stud[] ){**

**return stud[ 0 ].marks + stud[ 1 ].marks + stud[ 2 ].marks + stud[ 3 ].marks ;**

**}**

**int main( ){**

**Student stud[ 4 ] ;**

**stud[ 0 ].assign( "Sam", 23, 2002, 'A', 67.87 ) ;**

**stud[ 1 ].assign( "Dam", 26, 2001, 'B', 45.43 ) ;**

**stud[ 2 ].assign( "Lam", 21, 2005, 'C', 12.94 ) ;**

**stud[ 3 ].assign( "Tan", 25, 2003, 'D', 54.23 ) ;**

**stud[ 0 ].show() ;**

**stud[ 1 ].show() ;**

**stud[ 2 ].show() ;**

**stud[ 3 ].show() ;**

**cout << "Total Marks.: " << showM( stud ) << endl ;**

**return 0;**

**}**

**Program 9.1b**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 9.2**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class Area{**

**float area ;**

**public:**

**Area( float l, float b ){**

**area = l \* b ;**

**}**

**Area( float side ){**

**area = side \* side ;**

**}**

**float show( void ){**

**return area ;**

**}**

**};**

**int main( ){**

**Area rect( 4.2, 5.4 ) ;**

**Area sqr( 5.1 ) ;**

**cout << "\nArea of rect is.: " << rect.show( ) << endl ;**

**cout << "Area of sqr is.: " << sqr.show( ) << endl ;**

**return 0;**

**}**

**Program 9.2**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 9.3**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class A {**

**static int cnt ;**

**public:**

**static void counter( void ){**

**cnt ++ ;**

**}**

**static int show( void ){**

**return cnt ;**

**}**

**};**

**// Initializing the static data member**

**int A :: cnt = 0 ;**

**int main( ){**

**A obj1, obj2, obj3 ;**

**obj1.counter( ) ;**

**obj2.counter( ) ;**

**obj3.counter( ) ;**

**obj1.counter( ) ;**

**cout << "\nFunction is Called " << obj1.show( ) << " times" << endl ;**

**return 0;**

**}**

**Program 9.3**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 9.4**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class Student {**

**string name ;**

**short age ;**

**float marks ;**

**public:**

**friend void Make( Student \*, string, short, float ) ;**

**friend void show( Student ) ;**

**friend float showM( Student \*, int ) ;**

**};**

**void Make( Student \*obj, string name, short age, float marks ){**

**obj-> name = name ;**

**obj-> age = age ;**

**obj-> marks = marks ;**

**}**

**void show( Student obj ){**

**cout << "Name.: " << obj.name << endl ;**

**cout << "Age.: " << obj.age << endl ;**

**cout << "Marks.: " << obj.marks << endl ;**

**cout << endl ;**

**}**

**float showM( Student stud[], int n ){**

**float ans = 0.0f ;**

**for( int i = 0; i < n; i++ ){**

**ans += stud[ i ].marks ;**

**}**

**return ans ;**

**}**

**int main( ){**

**Student stud[ 3 ] ;**

**Make( &stud[ 0 ], "Sam", 23, 67.87 ) ;**

**Make( &stud[ 1 ], "Dam", 26, 45.43 ) ;**

**Make( &stud[ 2 ], "Lam", 21, 12.94 ) ;**

**show( stud[ 0 ] ) ;**

**show( stud[ 1 ] ) ;**

**show( stud[ 2 ] ) ;**

**cout << "Total Marks.: " << showM( stud, 3 ) << endl ;**

**return 0;**

**}**

**Program 9.4**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 9.5**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class Base {**

**public:**

**int var ;**

**};**

**class Child : Base{**

**int var2 ;**

**};**

**struct SBase {**

**public:**

**int var ;**

**};**

**struct SChild : Base{**

**int var2 ;**

**};**

**int main( ){**

**// Child obj ;**

**// obj.var2 = 5 ;**

**struct SChild obj2 ;**

**obj2.var2 = 5 ;**

**cout << obj2.var2 << endl ;**

**return 0;**

**}**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Diff. b/w Classes and Structure \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**=> By default Structure data members are public, whereas**

**Class data members are private by default**

**=> Inheritence by default in Classes is private, whereas**

**Inheritence by default in Structure is public**

**=> Structure uses less memory as compared to Classes because**

**=> You cannot change the default constructor of structure, whereas**

**You can change class default constructor.**

**=> In structure, all the value types are allocated on stack, whereas**

**In class, all value types are allocated on heap.**

**\*/**

**Program 9.5**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 9.6**

**Source Code:**

**#include<iostream>**

**#include<iomanip>**

**using namespace std;**

**int main()**

**{**

**cout<<"enter our choice"<<endl;**

**cout<<"1.come to next line \n2.setting field with \n3.fill string with \* after setw(15) function" <<endl;**

**int n;**

**long str=123456789;**

**cout<<"before before any operation "<<str;**

**while(1){**

**cout<<endl;**

**cin>>n;**

**switch(n)**

**{**

**case 1 :**

**cout<<"before going to next line "<<str;**

**cout<<endl;**

**cout<<str;**

**break;**

**case 2:**

**cout<<"after setting field width"<<str<<endl;**

**cout<<setw(15);**

**cout<<str;**

**break;**

**case 3:**

**cout<<"before setfill "<<str<<endl;**

**cout<<setfill('\*')<<setw(15);**

**cout<<str;**

**break;**

**default : return 0;**

**}**

**}**

**return 0 ;**

**}**

**Program 9.6**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 10.2**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**class student{**

**string name;**

**string add;**

**int roll\_no;**

**char sec;**

**public:**

**void set\_values(){**

**getline(cin,name);**

**getline(cin,add);**

**cin>>roll\_no;**

**cin>>sec;**

**}**

**void show\_data( ){**

**cout << name << " " << add << " " << roll\_no << " " << sec ;**

**}**

**};**

**int main(){**

**student \*ptr;**

**student obj1;**

**ptr=&obj1;**

**ptr->set\_values();**

**ptr->show\_data();**

**return 0;**

**}**

**Program 10.2**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence

**Program 10.3**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**class rem{**

**int a,b;**

**public:**

**rem(int a,int b){**

**this-> a = a ;**

**this-> b = b ;**

**cout << "\nans.: " << this-> a % this-> b << endl ;**

**}**

**};**

**int main(){**

**rem obj( 14, 3 ) ;**

**return 0;**

**}**

**Program 10.3**

**Output:**

A screenshot of a computer

Description automatically generated

**Program 10.4a**

**Source Code:**

**#include <iostream>**

**using namespace std ;**

**class A{**

**int data ;**

**public:**

**A( int data ){**

**this-> data = data ;**

**}**

**friend bool operator <( A, A ) ;**

**};**

**bool operator <( A obj1, A obj2 ){**

**return (bool) obj1.data < obj2.data ;**

**}**

**int main( ){**

**A obj1( 5 ), obj2( 10 ) ;**

**cout << ( obj1 < obj2 ) << endl ;**

**return 0;**

**}**

**Program 10.4a**

**Output:**

A screenshot of a computer

Description automatically generated

**Program 10.4b**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**class A{**

**int num;**

**public:**

**A(int x){**

**this-> num = x;**

**}**

**void operator +( A obj ){**

**cout << "Sum.: " << num +obj.num << endl ;**

**}**

**};**

**int main(){**

**A obj1(10);**

**A obj2(20);**

**obj1 +obj2;**

**return 0;**

**}**

**Program 10.4b**

**Output:**

A screenshot of a computer

Description automatically generated with medium confidence