|  |  |
| --- | --- |
| 1. | **Develop** a c++ program for default arguments |
| Sample Input | Enter the value: 10 15 25 30 |
| Sample Output | 80 |
| Test Case | a. 10,15 |
| b. 10,15,25 |
| c. 30,40,A |
| d. #,0.A,89 |
| e. 10, 15, 25, 80 |

|  |  |
| --- | --- |
| 2. | Develop a c++ program for adding the number using function overloading concept |
| Sample Input | Enter the value for a, b, c : 10 20 60 |
| Sample Output | The value of addition using two parameter is 30  The value of addition using three parameter is 60 |
| Test Case | 1.(25,&,09) |
| 2. (AB,78,09) |
| 3. (45,65,52) |
| 4. (0,0,%) |
| 5. (45,-6,2) |

|  |  |
| --- | --- |
| 3. | Declare a class box, with length(Public variable) and width(Private variable) use set width ()and get width() function to set the width and print the length and width .. |
| Sample Input | Enter the Length of box :6  Enter the Width of box :9 |
| Sample Output | Length of box : 6  Width of box : 9 |
| Test Case | 1.(10,8) |
| 2. (8,-6) |
| 3. (A,8) |
| 4. (@,%) |
| 5. (AB,0) |

|  |  |
| --- | --- |
| 4. | **Develop** a c++ program for matrix multiplication using **arrays** |
| Sample Input | enter the number of row=3  enter the number of column=3  enter the first matrix element=  1 2 3  1 2 3  1 2 3  enter the second matrix element=  1 1 1  2 1 2  3 2 1 |
| Sample Output | multiply of the matrix=  14 9 8  14 9 8  14 9 8 |
| Test Case | 1.(row=1 , column=2) |
| 2. (row=0 , column=3) |
| 3. (row=10 , column=A) |
| 4. (row=2 , column=2) |
| 5. (row=# , column=$) |

|  |  |
| --- | --- |
| 5. | **Develop** a simple program for **static** field to count the number of object created using C++. |
| Sample Input | Enter the Id of the Car: 101  Enter the name of the Car: Ferrari  Number of the Marks (1 - 10): 10  Enter the Id of the Car: 205  Enter the name of the Car: Mercedes  Number of the Marks (1 - 10): 9 |
| Sample Output | Id of the Car: 101  Name of the Car: Ferrari  Marks: 10  Id of the Car: 205  Name of the Car: Mercedes  Marks: 9  No. of objects created in the class: 2 |
| Test Case | 1.(1011,Ambassdor,-96) |
| 2. (101AB,Ambassdor,8) |
| 3. (#$,Ambassdor,#) |
| 4. (-52,Ambassdor,A) |
| 5. (101,Ambassdor,5) |

|  |  |
| --- | --- |
| 6. | **Develop** a C++ program to perform different arithmetic operations such as addition, subtraction, division, modulus and multiplication switch case |
| Sample Input | Calculator:  1.Addition  2. Subtraction.  3.Multiplication  4. Division  5.Modulus  Enter your choice:1  Number 1: 20  Number 2:30 |
| Sample Output | The value of addition is 50 |
| Test Case | 1.-23,90 |
| 2.SD,78 |
| 3.%$,76 |
| 4.45,34 |
| 5.0,A |

|  |  |
| --- | --- |
| 7. | **Develop** a Employee class with Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary. |
| Sample Input | Enter Name of the Employee : Suresh  Enter Address of the Employee : Vetri Nagar  Enter ID of the Employee :7001  Enter Mobile Number : 9898989898  Enter E-Mail ID of the Employee : aff@gmail.com  ENTER THE BASIC PAY OF THE PROGRAMMER => 80000 |
| Sample Output | =======================  PROGRAMMER PAYMENT SLIP  =======================  BASIC PAY => 80000  DEARNESS ALLOWANCE => 77600  HOUSE RENT ALLOWENCE => 8000  PROVIDENT FUND => 9600  CLUB FUND => 800  GROSS PAY => 175200  NET PAY => 164800 |
| Test Case | 1.(Sundar,Anna nagar,9000,9896532521,as@gmail.com,90000) |
| 2.(Samuel,Anna nagar,AB90,9896532521,as@gmail.com,90000) |
| 3.(Sivanesh,00000,8000,98965325UI,as@gmail.com,90000) |
| 4.(#$%,Nolambur,98965325UI,as@gmail.com,90000) |
| 5.(0000,00000,8000,98965325UI,as@gmail.com,00000) |

|  |  |
| --- | --- |
| 8. | Write a c++ program to remove duplicates from the sorted **array** |
| Sample Input | Sample Input:  Array = {15, 14, 25, 14, 32, 14, 31} |
| Sample Output | Sample Output:  Sorted Array = {14, 15, 25, 31, 32} |
| Test Case | 1. {16, 16, 16 16, 16} |
| 1. {0, 0, 0, 0} |
| 1. {-12, -78, -35, -42} |
| 1. {1,2,3,7,8,9,4,5,6} |
| 1. {1-2,2-3,3-4,4-5,5-6} |

|  |  |
| --- | --- |
| 9. | Write a program to enter the marks of a student in four subjects. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is 60>= and <75, then the grade is First Division. If aggregate is 50 >= and <60, then the grade is Second Division. If aggregate is 40>= and <50, then the grade is Third Division. Else the grade is Fail. |
| Sample Input | Please Enter the marks of five subjects:  50 50 50 50 50 |
| Sample Output | Total Marks = 250  Average Marks = 50  Marks Percentage = 50  Grade E |
| Test Case | 1. (89,58,63,0,-1) |
| 2. (25,35,63,0,1.03) |
| 3. (76,58,85,95) |
| 4. (89,58,$,0,-1) |
| 5. (AB,58,63,0,-1) |

|  |  |
| --- | --- |
| 10. | **Develop** a largest class with a,b, and m as member. Use setdata () for setting the data and friend void find\_Max (largest) function for finding the largest number. |
| Sample Input | Enter the first no: 52  Enter the second no: 63 |
| Sample Output | Maximum no is 63 |
| Test Case | 1. (0,0) |
| 2. (80,90) |
| 3. (@,$) |
| 4. (A,42) |
| 5. (-52,30) |