

DEPARTMENT OF MATHEMATICS  
PATTAMUNDAI COLLEGE, PATTAMUNDAI

**DSE-1 : PROGRAMMING IN C++**

**Section-A**

1. What is the full form of OOPS?
2. What is a class?
3. What is an object?
4. List the types of inheritance supported in C++.
5. What is the role of protected access specifier?
6. What is encapsulation?
7. What is abstraction?
8. What is inheritance?
9. Explain the purpose of the keyword volatile.
10. What is an inline function?
11. What is a storage class?
12. Mention the storage classes names in C++.
13. What is the role of mutable storage class specifier?
14. Distinguish between shallow copy and deep copy.
15. What is a pure virtual function?
16. What is an abstract class in C++?
17. What is a reference variable in C++?
18. What is role of static keyword on class member variable?
19. Explain the static member function.
20. Name the data type which can be used to store wide characters in C++.
21. What are/is the operator/operators used to access the class members?
22. Can we initialize a class/structure member variable as soon as the same is defined?
23. What is the data type to store the Boolean value?
24. What is function overloading?
25. What is operator overloading?
26. Do we have a String primitive data type in C++?
27. Name the default standard streams in C++.

28. Which access specifier/s can help to achieve data hiding in C++?
29. When a class member is defined outside the class, which operator can be used to associate the function definition to a particular class?
30. What is a destructor? Can it be overloaded?
31. What is a constructor?
32. What is a default constructor? Can we provide one for our class?
33. Which operator can be used in C++ to allocate dynamic memory?
34. What is the purpose of 'delete' operator?
35. Can I use malloc() function of C language to allocate dynamic memory in C++?
36. Can I use 'delete' operator to release the memory which was allocated using malloc() function of C language?
37. What is a friend function?
38. What is a copy constructor?
39. Does C++ support exception handling? If so what are the keywords involved in achieving the same.
40. Explain the pointer – this.
41. What is the difference between the keywords struct and class in C++?
42. Can we implement all the concepts of OOPS using the keyword struct?
43. What is the block scope variable in C++?
44. What is the role of the file opening mode ios::trunc?
45. What is the scope resolution operator?
46. What is a namespace?
47. What are command line arguments?
48. What is a class template?
49. How can we catch all kind of exceptions in a single catch block?
50. What is keyword auto for?
51. What is a static variable?
52. What is the purpose of extern storage specifier.
53. What is the meaning of base address of the array?
54. When should we use the register storage specifier?
55. Can a program be compiled without main() function?
56. Where an automatic variable is stored?
57. What is a container class?
58. What is a token?

59. What is a preprocessor?
60. What are command line arguments?
61. What is remainder for  $5.0 \% 2$ ?
62. Which compiler switch to be used for compiling the programs using math library with g++ compiler?
63. Can we resize the allocated memory which was allocated using 'new' operator?
64. Who designed C++ programming language?
65. Which operator can be used to determine the size of a data type/class or variable/object?
66. How can we refer to the global variable if the local and the global variable names are same?
67. What are valid operations on pointers?
68. What is recursion?
69. What is the first string in the argument vector w.r.t command line arguments?
70. What is the maximum length of an identifier?
71. What is the default function call method?
72. What are available mode of inheritance to inherit one class from another?
73. What is the difference between delete and delete[]?
74. Does an abstract class in C++ need to hold all pure virtual functions?
75. Is it legal to assign a base class object to a derived class pointer?
76. What happens if an exception is thrown outside a try block?
77. Are the exceptions and error same?
78. What is function overriding?
79. Which function is used to move the stream pointer for the purpose of reading data from stream?
80. Which function is used to move the stream pointer for the purpose of writing data from stream?
81. Are class functions taken into consideration as part of the object size?
82. Can we create an empty class? If so what would be the size of such object.
83. What is 'std'?
84. What is the full form of STL?
85. What is 'cout'?
86. What is 'cin'?
87. What is the use of the keyword 'using'?

- 88.If a pointer declared for a class, which operator can be used to access its class members?
- 89.What is difference between including the header file with-in angular braces < > and double quotes “ “?
- 90.S++ or S = S+1, which can be recommended to increment the value by 1 and why?
- 91.What is the difference between actual and formal parameters?
- 92.What is the difference between variable declaration and variable definition?
- 93.Declaration associates type to the variable whereas definition gives the value to the variable.
- 94.Which key word is used to perform unconditional branching?
- 95.Is 068 a valid octal number?
- 96.What is the purpose of #undef preprocessor?
- 97.Can we nest multi line comments in a C++ code?
- 98.What is a virtual destructor?
- 99.What is the order of objects destroyed in the memory?
- 100.What is a friend class?

### **Section-B**

- 1.Write any four important characteristics of object oriented programming. Give example of any one of the characteristics using C++.
- 2.Explain data hiding with an example.
- 3.What is function overloading? Give an example in C++ to illustrate function overloading.  
or What do you understand by function overloading? Give an example illustrating its use in a C++ program.
- 4.Write the output of the following C++ code. Also, write the name of feature of Object Oriented Programming used in the following program jointly illustrated by the functions [I] to [IV].

```

#include<iostream.h>
void Print() //Function[I]
{
for(int K=1; K<=60; K++)
cout<<"-";
cout<<endl;
}
void Print(int N) //Function[II]
{
for(int K=1; K<=N; K++)
cout<<"*";
cout<<endl;
}
void Print(int A, int B)//Function[III]
}
for(int K=1; K<=B; K++)
cout<<A*K;
cout<<endl;
}
void Print(char T, int N)//Function[IV]
{
for(int K=1; K<=N; K++)
cout<<T;
cout<<endl;
}
void main()
{
int U=9, V=4, W=3;
char C= '@';
Print(C, V);
Print(U, W);
}

```

5. Write the output of the following C++ code. Also, write the name of feature of Object Oriented Programming used in the following program jointly illustrated by the functions [I] to [IV]:

```

#include<iostream.h>
void Lined //Function[I]
{
for(int L=1;L<=80;L++)
cout<<"-";
cout<<endl;
}
void Line(int N) //Function[II]
{
for(int L=1;L<=N;L++)
cout<<"*";
cout<<endl;
}

```

```

void Line(char C,int N) //Function[III]
{
for(int L=1:L<=N:L++)
cout<<C;
cout<<endl;
}
void Line(int M, int N)//Function[IV]
{
for(int L=1:L<N:L++)
cout<<M*L;
cout<<endl;
}
void main()
{
int A=9, B=4, C=3;
char K='#':
Line (K,B);
Line (A,C);
}

```

6.What do you understand by Data Encapsulation and Data Hiding? Also, give an example in c++.

7.What do you understand by polymorphism? Give an example illustrating its use in a C++ program.

8.How are abstraction and encapsulation interrelated?

9.What is event driven programming?.

10.What is the significance of classes in OOPs?.

11.What are the advantages of object oriented programming over procedural oriented programming?

12.Encapsulation is one of the major properties of OOP. How is it implemented in C++?

OR Define the term data encapsulation in terms of object oriented programming. Give a suitable example using a C++ code to illustrate the same.

13.What is operator overloading? Explain with example

## Section -C

1. C++ Program to Print Number Entered by User
2. C++ Program to Add Two Numbers
3. C++ Program to Find Quotient and Remainder
4. C++ Program to Find Size of int, float, double and char in Your System
5. C++ Program to Swap Two Numbers
6. C++ Program to Check Whether Number is Even or Odd
7. C++ Program to Check Whether a character is Vowel or Consonant.
8. C++ Program to Find Largest Number Among Three Numbers
9. C++ Program to Find All Roots of a Quadratic Equation
10. C++ Program to Calculate Sum of Natural Numbers
11. C++ Program to Check Leap Year
12. C++ Program to Find Factorial
13. C++ Program to Generate Multiplication Table
14. C++ Program to Display Fibonacci Series
15. C++ Program to Find GCD
16. C++ Program to Find LCM
17. C++ Program to Reverse a Number
18. C++ Program to Calculate Power of a Number
19. Increment ++ and Decrement -- Operator Overloading in C++ Programming
20. C++ Program to Subtract Complex Number Using Operator Overloading
21. C++ Program to Find ASCII Value of a Character
22. C++ Program to Multiply two Numbers
23. C++ Program to Check Whether a Number is Palindrome or Not
24. C++ Program to Check Whether a Number is Prime or Not
25. C++ Program to Display Prime Numbers Between Two Intervals
26. C++ Program to Check Armstrong Number
27. C++ Program to Display Armstrong Number Between Two Intervals
28. C++ Program to Display Factors of a Number
29. C++ Programs To Create Pyramid and Pattern
30. C++ Program to Make a Simple Calculator to Add, Subtract, Multiply or Divide Using switch...case

31. C++ Program to Display Prime Numbers Between Two Intervals Using Functions

---

32. C++ Program to Check Prime Number By Creating a Function

---

33. C++ Program to Check Whether a Number can be Express as Sum of Two Prime Numbers

34. C++ program to Find Sum of Natural Numbers using Recursion

35. C++ program to Calculate Factorial of a Number Using Recursion

36. C++ Program to Find G.C.D Using Recursion

37. C++ Program to Convert Binary Number to Decimal and vice-versa

38. C++ Program to Convert Octal Number to Decimal and vice-versa

39. C++ Program to Convert Binary Number to Octal and vice-versa

40. C++ program to Reverse a Sentence Using Recursion

41. C++ Program to Calculate Power Using Recursion

42. C++ Program to Calculate Average of Numbers Using Arrays

43. C++ Program to Find Largest Element of an Array

44. C++ Program to Calculate Standard Deviation

45. C++ Program to Add Two Matrix Using Multi-dimensional Arrays

46. C++ Program to Multiply Two Matrix Using Multi-dimensional Arrays

47. C++ Program to Find Transpose of a Matrix

48. C++ Program to Multiply two Matrices by Passing Matrix to Function

49. C++ Program to Access Elements of an Array Using Pointer

50. C++ Program to Swap Numbers in Cyclic Order Using Call by Reference

51. C++ Program to Find the Frequency of Characters in a String

52. C++ Program to Find the Number of Vowels, Consonants, Digits and White Spaces in a String

53. C++ Program to Remove all Characters in a String Except Alphabets.

54. C++ Program to Find the Length of a String

55. C++ Program to Concatenate Two Strings

56. C++ Program to Copy Strings

57. C++ Program to Sort Elements in Lexicographical Order (Dictionary Order)

58. C++ Program to Store Information of a Student in a Structure

59. C++ Program to Add Two Distances (in inch-feet) System Using Structures

60. C++ Program to Add Complex Numbers by Passing Structure to a Function

## 61. C++ Program to Calculate Difference Between Two Time Period

---