YENEPOYA
(DEEMED TO BE UNIVERSITY)
Recognized under Sec 3(A) of the UGC Act 1956
Accredited by NAAC with 'A' Grade

THE YENEPOYA INSTITUTE OF ARTS, SCIENCE, COMMERCE AND MANAGEMENT

A Constituent unit of Yenepoya (Deemed to be University)

Question Bank

II Semester BCA (Big Data Analytics & Cloud Computing)

BCS201 - C++ PROGRAMMING

Section A

1. Who invented C++?
   a) Dennis Ritchie
   b) Ken Thompson
   c) Brian Kernighan
   d) Bjarne Stroustrup
2. What is C++?
   a) C++ is an object oriented programming language
   b) C++ is a procedural programming language
   c) C++ supports both procedural and object oriented programming language
   d) C++ is a functional programming language
3. Which of the following is the correct syntax of including a user defined header files in C++?
   a) #include [userdefined]
   b) #include “userdefined”
   c) #include <userdefined.h>
   d) #include <userdefined>
4. Which of the following is used for comments in C++?
   a) /* comment */
   b) // comment */
   c) // comment
   d) both // comment or /* comment */
5. Which of the following user-defined header file extension used in c++?
   a) hg
   b) cpp
   c) h
   d) hf
6. Which of the following is a correct identifier in C++?
   a) VAR_1234
   b) $var_name
   c) 7VARNAME
   d) 7var_name
7. Which of the following is not a type of Constructor in C++?
   a) Default constructor
   b) Parameterized constructor

c) Copy constructor
d) Friend constructor

8. Which of the following approach is used by C++?
   a) Left-right
   b) Right-left
   c) Bottom-up
   d) Top-down
9. What is virtual inheritance in C++?
   a) C++ technique to enhance multiple inheritance
   b) C++ technique to ensure that a private member of the base class can be accessed somehow
   c) C++ technique to avoid multiple inheritances of classes
   d) C++ technique to avoid multiple copies of the base class into children/derived class
10. What happens if the following C++ statement is compiled and executed?

```
int *ptr = NULL;
delete ptr;
```

   a) The program is not semantically correct
   b) The program is compiled and executed successfully
   c) The program gives a compile-time error
   d) The program compiled successfully but throws an error during run-time

11. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
using namespace std;
int main(int argc, char const *argv[])
{
        char s1[6] = "Hello";
        char s2[6] = "World";
        char s3[12] = s1 + " " + s2;
        cout<<s3;
        return 0;
}
```

a) Hello
b) World
c) Error
d) Hello World

12. What is the difference between delete and delete[] in C++?
a) delete is syntactically correct but delete[] is wrong and hence will give an error if used in any case
b) delete is used to delete normal objects whereas delete[] is used to pointer objects
c) delete is a keyword whereas delete[] is an identifier
d) delete is used to delete single object whereas delete[] is used to multiple(array/pointer of) objects

13. Which of the following is correct about this pointer in C++?
a) this pointer is passed as a hidden argument in all static variables of a class
b) this pointer is passed as a hidden argument in all the functions of a class

c) this pointer is passed as a hidden argument in all non-static functions of a class
d) this pointer is passed as a hidden argument in all static functions of a class

14. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
#include <algorithm>
using namespace std;
int main()
{
    string s = "spaces in text";
    s.erase(remove(s.begin(), s.end(), ' ' ), s.end() ) ;
    cout << s << endl;
 }
```

a) spacesintext
b) spaces in text
c) spaces
d) spaces in

15. Which of the following type is provided by C++ but not C?
a) double
b) float
c) int
d) bool

16. Which of the following correctly declares an array in C++?
a) array{10};
b) array array[10];
c) int array;
d) int array[10];

17. Which is more effective while calling the C++ functions?
a) call by object
b) call by pointer
c) call by value
d) call by reference

18. What will be the output of the following C++ program?

```
#include <iostream>
#include <string>
#include <cstring>
using namespace std;
int main(int argc, char const *argv[])
{
        const char *a = "Hello\0World";
        cout<<a;
        return 0;
}
```

a) Hello
b) World

c) Error
d) Hello World

19. Which of the following is used to terminate the function declaration in C++?
a) ;
b) ]
c) )
d) :

20.What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    char c = 74;
    cout << c;
    return 0;
}
```

a) I
b) J
c) A
d) N

21. What is Inheritance in C++?
a) Deriving new classes from existing classes
b) Overloading of classes
c) Classes with same names
d) Wrapping of data into a single class

22. Which of the following symbol is used to declare the preprocessor directives in C++?
a) $
b) ^
c) #
d) *

23. What is meant by a polymorphism in C++?
a) class having only single form
b) class having four forms
c) class having many forms
d) class having two forms

24. What is abstract class in C++?
a) Any Class in C++ is an abstract class
b) Class from which any class is derived
c) Class specifically used as a base class with atleast one virtual functions
d) Class specifically used as a base class with atleast one pure virtual functions

25. Which concept allows you to reuse the written code in C++?
a) Inheritance
b) Polymorphism
c) Abstraction
d) Encapsulation

26. How structures and classes in C++ differ?
a) Structures by default hide every member whereas classes do not
b) In Structures, members are public by default whereas, in Classes, they are private by default
c) Structures cannot have private members whereas classes can have
d) In Structures, members are private by default whereas, in Classes, they are public by default

27. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main ()
{
    int a, b, c;
    a = 2;
    b = 7;
    c = (a > b) ? a : b;
    cout << c;
     return 0;
 }
```

a) 12
b) 14
c) 6
d) 7

28. What will be the output of the following C++ code snippet?

```
1.  #include <stdio.h>
2.  #include<iostream>
3.  using namespace std;
4.  int main ()
5.  {
6.      int array[] = {0, 2, 4, 6, 7, 5, 3};
7.      int n, result = 0;
8.      for (n = 0; n < 8; n++)
9.      {
10.         result += array[n];
11.      }
12.      cout << result;
13.      return 0;
14.  }
```

a) 21
b) 27
c) 26
d) 25

29. Pick the odd one out.
a) array type
b) character type
c) boolean type
d) integer type

30. Which data type is used to represent the absence of parameters?
a) int
b) short
c) void
d) float

31. What does '\a' escape code represent?
a) alert
b) backslash
c) tab
d) form feed

32. Which type is best suited to represent the logical values?
a) integer
b) boolean
c) character
d) float

33. Identify the user-defined types from the following?
a) enumeration
b) classes
c) both enumeration and classes
d) int

34. Which of the following statements are true?

```
int f(float)
```

a) f is a function taking an argument of type int and returning a floating point number
b) f is a function taking an argument of type float and returning an integer
c) f is a function of type float
d) f is a function of type int

35. The value 132.54 can be represented using which data type?
a) double
b) void
c) int
d) bool

36. Pick the odd one out.
a) integer, character, boolean, floating
b) enumeration, classes
c) integer, enum, void
d) arrays, pointer, classes

36. What happens when a null pointer is converted into bool?
a) an error is flagged
b) bool value evaluates to true
c) bool value evaluates to false
d) the statement is ignored

37. Which of the following statements are false?
a) bool can have two values and can be used to express logical expressions

b) bool cannot be used as the type of the result of the function
c) bool can be converted into integers implicitly
d) a bool value can be used in arithmetic expressions

38. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int f(int p, int q)
{
    if (p > q)
        return p;
    else
        return q;
}
 main()
 {
     int a = 5, b = 10;
     int k;
     bool x = true;
     bool y = f(a, b);
     k =((a * b) + (x + y));
     cout << k;
 }
```

a) 55
b) 62
c) 52
d) 75

39. What is the value of p in the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int p;
    bool a = true;
    bool b = false;
    int x = 10;
    int y = 5;
     p = ((x | y) + (a + b));
     cout << p;
     return 0;
 }
```

a) 0
b) 16
c) 12
d) 2

40. How many characters are specified in the ASCII scheme?
a) 64
b) 128
c) 256
d) 24

41. The size_t integer type in C++ is?
a) Unsigned integer of at least 64 bits
b) Signed integer of at least 16 bits
c) Unsigned integer of at least 16 bits
d) Signed integer of at least 64 bits

42. . What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
        int x = -1;
    unsigned int y = 2;

    if(x > y)
    {
        cout << "x is greater";
        }
     else
     {
                cout << "y is greater";
        }
 }
```

a) x is greater
b) y is greater
c) implementation defined
d) arbitrary

43. Which of the following is not one of the sizes of the floating point types?
a) short float
b) float
c) long double
d) double

44. What will be the output of the following C++ code?

```
#include <iostream>
  using namespace std;
  int main()
  {
     float num1 = 1.1;
     double num2 = 1.1;
     if (num1 == num2)
        cout << "stanford";
     else
         cout << "harvard";
      return 0;
   }
```

a) harvard
b) stanford
c) compile time error
d) runtime error

45. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    float i = 123.0f;
    cout << i << endl;
    return 0;
}
```

a) 123.00
b) 1.23
c) 123
d) compile time error

46. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    float f1 = 0.5;
    double f2 = 0.5;
    if (f1 == 0.5f)
        cout << "equal";
    else
        cout << "not equal";
    return 0;
}
```

a) equal
b) not equal
c) compile time error
d) runtime error

47. Which is correct with respect to the size of the data types?
a) char > int < float
b) int < char > float
c) char < int < float
d) char < int < double

48. The size of an object or a type can be determined using which operator?
a) malloc
b) sizeof
c) malloc
d) calloc

49. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main ( )
{
    static double i;
```

```
    i = 20;
     cout << sizeof(i);
  return 0;
 }
```

a) 4
b) 2
c) 8
d) error

50. What will be the output of the following C++ code (in 32-bit systems)?

```
#include <iostream>
using namespace std;
int main()
{
    cout << sizeof(char);
    cout << sizeof(int);
    cout << sizeof(float);
    return 0;
}
```

a) 1 4 4
b) 1 4 8
c) 1 8 8
d) 1 8 2

51. The operator used for dereferencing or indirection is ____
a) *
b) &
c) ->
d) –>>

52. Choose the right option.

string* x, y;

a) x is a pointer to a string, y is a string
b) y is a pointer to a string, x is a string
c) both x and y are pointers to string types
d) y is a pointer to a string

53. Which one of the following is not a possible state for a pointer.
a) hold the address of the specific object
b) point one past the end of an object
c) zero
d) point to a type

54. What will happen in the following C++ code snippet?

```
int a = 100, b = 200;
int *p = &a, *q = &b;
p = q;
```

a) b is assigned to a
b) p now points to b

c) a is assigned to b
d) q now points to a

55. The correct statement for a function that takes pointer to a float, a pointer to a pointer to a char and returns a pointer to a pointer to a integer is _____
a) int **fun(float**, char**)
b) int *fun(float*, char*)
c) int **fun(float*, char**)
d) int ***fun(*float, **char)

56. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    char arr[20];
    int i;
    for(i = 0; i < 10; i++)
        *(arr + i) = 65 + i;
    *(arr + i) = '\0';
     cout << arr;
     return(0);
 }
```

a) ABCDEFGHIJ
b) AAAAAAAAAA
c) JJJJJJJJ
d) AAAAAAJJJJ

57. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    char *ptr;
    char Str[] = "abcdefg";
    ptr = Str;
    ptr += 5;
    cout << ptr;
     return 0;
 }
```

a) fg
b) cdef
c) defg
d) abcd

58. Which of the following correctly declares an array?
a) int array[10];
b) int array;
c) array{10};
d) array array[10];

59. What is the index number of the last element of an array with 9 elements?
a) 9
b) 8
c) 0
d) Programmer-defined

60. What is the correct definition of an array?
a) An array is a series of elements of the same type in contiguous memory locations
b) An array is a series of element
c) An array is a series of elements of the same type placed in non-contiguous memory locations
d) An array is an element of the different type

61. Which of the following accesses the seventh element stored in array?
a) array[6];
b) array[7];
c) array(7);
d) array;

62. What will be the output of the following C++ code?

```
#include <stdio.h>
#include<iostream>
using namespace std;
int array1[] = {1200, 200, 2300, 1230, 1543};
int array2[] = {12, 14, 16, 18, 20};
int temp, result = 0;
int main()
{
    for (temp = 0; temp < 5; temp++)
    {
        result += array1[temp];
    }
    for (temp = 0; temp < 4; temp++)
    {
        result += array2[temp];
    }
    cout << result;
return 0;
 }
```

a) 6553
b) 6533
c) 6522
d) 12200

63. What will be the output of the following C++ code?

```
#include <stdio.h>
#include <iostream>
using namespace std;
int main()
{
    char str[5] = "ABC";
    cout << str[3];
```

```
    cout << str;
    return 0;
  }
```

a) ABC
b) ABCD
c) AB
d) AC

64. What is the meaning of the following declaration?

int(*p[5])();

a) p is pointer to function
b) p is array of pointer to function
c) p is pointer to such function which return type is the array
d) p is pointer to array of function

65. What is size of generic pointer in C++ (in 32-bit platform)?
a) 2
b) 4
c) 8
d) 0

66. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int a[2][4] = {3, 6, 9, 12, 15, 18, 21, 24};
    cout << *(a[1] + 2) << *(*(a + 1) + 2) << 2[1[a]];
    return 0;
}
```

a) 15 18 21
b) 21 21 21
c) 24 24 24
d) Compile time error

67. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    const char *arr[] = {"C", "C++", "Java", "VBA"};
    const char *(*ptr)[4] = &arr;
    cout << ++(*ptr)[2];
     return 0;
}
```

a) ava
b) java

c) c++
d) compile time error

68. . What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int arr[] = {4, 5, 6, 7};
    int *p = (arr + 1);
    cout << *p;
    return 0;
}
```

a) 4
b) 5
c) 6
d) 7

69. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main ()
{
    int numbers[5];
    int * p;
    p = numbers;  *p = 10;
    p++;  *p = 20;
    p = &numbers[2];  *p = 30;
     p = numbers + 3;  *p = 40;
     p = numbers;  *(p + 4) = 50;
     for (int n = 0; n < 5; n++)
         cout << numbers[n] << ",";
     return 0;
 }
```

a) 10,20,30,40,50,
b) 1020304050
c) compile error
d) runtime error

70.  What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
     int arr[] = {4, 5, 6, 7};
     int *p = (arr + 1);
     cout << *arr + 9;
  return 0;
}
```

a) 12
b) 5
c) 13
d) error

71. The constants are also called as _____
a) const
b) preprocessor
c) literals
d) variables

72. What are the parts of the literal constants?
a) integer numerals
b) floating-point numerals
c) strings and boolean values
d) all of the mentioned

73. How are the constants declared?
a) const keyword
b) #define preprocessor
c) both const keyword and #define preprocessor
d) $define

74. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
#define PI 3.14159
int main ()
{
    float r = 2;
    float circle;
    circle = 2 * PI * r;
    cout << circle;
     return 0;
 }
```

a) 12.5664
b) 13.5664
c) 10
d) 15

75. Which of the following statement is not true about preprocessor directives?
a) These are lines read and processed by the preprocessor
b) They do not produce any code by themselves
c) These must be written on their own line
d) They end with a semicolon

76. Regarding the following statement which of the statements is true?

const int a = 100;

a) Declares a variable a with 100 as its initial value
b) Declares a construction a with 100 as its initial value

c) Declares a constant a whose value will be 100
d) Constructs an integer type variable with an as identifier and 100 as the value

77. The difference between x and ‘x’ is?
a) The first one refers to a variable whose identifier is x and the second one refers to the character constant x
b) The first one is a character constant x and the second one is the string literal x
c) Both are same
d) Both are string literal

78. Which value can we not assign to reference?
a) integer
b) floating
c) unsigned
d) null

79. Identify the incorrect statement.
a) Reference is the alternate name of the object
b) A reference value once defined can be reassigned
c) A reference value once defined cannot be reassigned
d) Reference is the alternate name of the variable

80. What are the references in C++?
a) An alternative name for already existing variables
b) A pointer to a variable
c) A new type of variables
d) A new type of constant variable

81. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
#include <cstdlib>

using namespace std;

int main(int argc, char const *argv[])
{
        int &q = 5;
        cout<<q;
        return 0;
}
```

a) 5
b) Run-time error
c) Segmentation fault
d) Compile-time error

82. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
#include <cstdlib>
```

```
using namespace std;

int main(int argc, char const *argv[])
{
        int &p;
        int a = 5;
        &p = a;
        cout<<p;
        return 0;
}
```

a) 5
b) 55
c) Error
d) Segmentation fault

83. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
#include <cstdlib>

using namespace std;

int main(int argc, char const *argv[])
{
        int a = 5;
        int &p = a;
        cout<<p;
        return 0;
}
```

a) 5
b) Run-time error
c) Segmentation fault
d) Compile-time error

84. What is the difference between references and pointers?
a) References are an alias for a variable whereas pointer stores the address of a variable
b) References and pointers are similar
c) References stores address of variables whereas pointer points to variables
d) Pointers are an alias for a variable whereas references stores the address of a variable

85. Pick the correct statement about references in C++.
a) References stores the address of variables
b) References and variables both have the same address
c) References use dereferencing operator(*) to access the value of variable its referencing
d) References were also available in C

86. What will be the output of the following C++ code?

```
#include <iostream>
#include <string>
```

```
#include <cstdlib>

using namespace std;

int main(int argc, char const *argv[])
{
        int a = 5;
        int *p = &a;
        int &q = a;
        cout<<p<<endl;
        cout<<q<<endl;
        return 0;
}
```

a) Address of a followed by 5 in next line
b) Address of p followed by 5 in next line
c) Address of a followed by Address of a in next line
d) Address of p followed by Address of q in next line

87. Pick the correct statement about references.
a) References can be assigned value NULL
b) References once assigned cannot be changed to refer another variable
c) Reference should not be initialized when created
d) Reference is the same as pointers

88. Which of the following operator is used while declaring references?
a) *
b) &
c) ^
d) ->

89. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int f(int &x, int c)
{
  c  = c - 1;
  if (c == 0) return 1;
  x = x + 1;
  return f(x, c) * x;
}
int main(int argc, char const *argv[])
{
        int a = 4;
        cout<<f(a,a);
        return 0;
}
```

**a)** 343
b) 336
c) 120
d) 840

90. Which of the following is incorrect?
a) References cannot be NULL
b) A reference must be initialized when declared
c) Once a reference is declared, it cannot be modified later to reference another object i.e. it cannot be reset
d) References cannot refer to a constant value

91. Which of the following function must use reference.
a) Assignment operator function
b) Copy Constructor
c) Destructor
d) Parameterized constructor

92. What will be the output of the following C++ code?

```
#include<iostream>
using namespace std;

int main()
{
  int x = 10;
  int& ref = x;
  ref = 20;
  cout << x << endl ;
  x = 30;
  cout << ref << endl;
  return 0;
}
```

a) 20
   30
b) 10
   10
c) 10
   20
d) 10
   30

93. How a reference is different from a pointer?
a) A reference cannot be null
b) A reference once established cannot be changed
c) The reference doesn't need an explicit dereferencing mechanism
d) All of the mentioned

94. Which of the following statement(s) is/are correct?
a) * operator is used to declare a reference
b) A reference variable defined to refer a particular variable can refer to any other variable also
c) References must always be initialized inside classes
d) A variable can have more than one references
95. The void pointer can point to which type of objects?
a) int
b) float
c) double
d) all of the mentioned

96. When does the void pointer can be dereferenced?
a) when it doesn't point to any value
b) when it cast to another type of object
c) using delete keyword
d) using shift keyword

97. The pointer can point to any variable that is not declared with which of these?
a) const
b) volatile
c) both const & volatile
d) static

98. A void pointer cannot point to which of these?
a) methods in c++
b) class member in c++
c) methods & class member in c++
d) none of the mentioned

99. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    char c;
    void *data;
    i = 2;
    c = 'd';
     data = &i;
     cout << "the data points to the integer value" << data;
     data = &c;
     cout << "the data now points to the character" << data;
     return 0;
 }
```

a) 2d
b) two memory addresses
c) 3d
d) 4d

100. What we can't do on a void pointer?
a) pointer arithmetic
b) pointer functions
c) pointer objects
d) pointer functions & objects

101. The data elements in the structure are also known as what?
a) objects
b) members
c) data
d) objects & data

102. What will be used when terminating a structure?
a) :
b) }
c) ;
d) ;;

103. What will happen when the structure is declared?
a) it will not allocate any memory
b) it will allocate the memory
c) it will be declared and initialized
d) it will be declared

104. The declaration of the structure is also called as?
a) structure creator
b) structure signifier
c) structure specifier
d) structure creator & signifier

105. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
struct Time
{
    int hours;
    int minutes;
    int seconds;
};
int toSeconds(Time now);
 int main()
 {
     Time t;
     t.hours = 5;
     t.minutes = 30;
     t.seconds = 45;
     cout << "Total seconds: " << toSeconds(t) << endl;
     return 0;
 }
 int toSeconds(Time now)
 {
     return 3600 * now.hours + 60 * now.minutes + now.seconds;
 }
```

a) 19845
b) 20000
c) 15000
d) 19844

106. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
```

```
struct ShoeType
{
   string style;
   double price;
};
  ShoeType shoe1, shoe2;
  shoe1.style = "Adidas";
  shoe1.price = 9.99;
  cout << shoe1.style << " $ "<< shoe1.price;
  shoe2 = shoe1;
  shoe2.price = shoe2.price / 9;
  cout << shoe2.style << " $ "<< shoe2.price;
  return 0;
}
```

a) Adidas $ 9.99Adidas $ 1.11
b) Adidas $ 9.99Adidas $ 9.11
c) Adidas $ 9.99Adidas $ 11.11
d) Adidas $ 11.11Adidas $ 11.11

107. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
struct sec
{
   int a;
   char b;
};
int main()
{
    struct sec s ={25,50};
    struct sec *ps =(struct sec *)&s;
    cout << ps->a << ps->b;
    return 0;
}
```

a) 252
b) 253
c) 254
d) 262

108. Which of the following is a properly defined structure?
a) struct {int a;}
b) struct a_struct {int a;}
c) struct a_struct int a;
d) struct a_struct {int a;};

109. Which of the following accesses a variable in structure *b?
a) b->var;
b) b.var;
c) b-var;
d) b>var;

110. Which operator is having the right to left associativity in the following?
a) Array subscripting

b) Function call
c) Addition and subtraction
d) Type cast

111. Which operator is having the highest precedence?
a) postfix
b) unary
c) shift
d) equality

112. What is this operator called ?:?
a) conditional
b) relational
c) casting operator
d) unrelational

113. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int a;
    a = 5 + 3 * 5;
    cout << a;
    return 0;
}
```

a) 35
b) 20
c) 25
d) 30

114. What is the use of dynamic_cast operator?
a) it converts virtual base class to derived class
b) it converts the virtual base object to derived objects
c) it will convert the operator based on precedence
d) it converts the virtual base object to derived class

115. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int a = 5, b = 6, c, d;
    c = a, b;
    d = (a, b);
    cout << c << ' ' << d;
    return 0;
 }
```

a) 5 6
b) 6 5

c) 6 7
d) 6 8

116. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int i, j;
    j = 10;
    i = (j++, j + 100, 999 + j);
    cout << i;
    return 0;
 }
```

a) 1000
b) 11
c) 1010
d) 1001

117. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main ()
{
    int x, y;
    x = 5;
    y = ++x * ++x;
    cout << x << y;
    x = 5;
     y = x++ * ++x;
     cout << x << y;
     return 0;
}
```

a) 749735
b) 736749
c) 367497
d) 367597

118. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int main()
{
    int a = 5, b = 6, c;
    c = (a > b) ? a : b;
    cout << c;
    return 0;
}
```

a) 6
b) 5

c) 4
d) 7

119. How are many sequences of statements present in c++?
a) 4
b) 3
c) 5
d) 6

120. The if..else statement can be replaced by which operator?
a) Bitwise operator
b) Conditional operator
c) Multiplicative operator
d) Addition operator

121. The switch statement is also called as?
a) choosing structure
b) selective structure
c) certain structure
d) bitwise structure

122. The destination statement for the goto label is identified by what label?
a) $
b) @
c) *
d) :

123. What will be the output of the following C++ code?

```
#include <iostream>
  using namespace std;
  int main ()
  {
     int n;
     for (n = 5; n > 0; n--)
     {
        cout << n;
        if (n == 3)
            break;
     }
     return 0;
  }
```

a) 543
b) 54
c) 5432
d) 53

124. What will be the output of the following C++ code?

```
  #include <iostream>
  using namespace std;
  int main()
  {
```

```
int i;
for (i = 0; i < 10; i++);
{
    cout << i;
}
 return 0;
}
```

a) 0123456789
b) 10
c) 012345678910
d) error

125. How many types of loops are there in C++?
a) 4
b) 2
c) 3
d) 1

126. Which looping process is best used when the number of iterations is known?
a) for
b) while
c) do-while
d) all looping processes require that the iterations be known

127. How many types of comments are there in c++?
a) 1
b) 2
c) 3
d) 4

128. What is a comment in c++?
a) comments are parts of the source code disregarded by the compiler
b) comments are executed by the compiler to find the meaning of the comment
c) comments are executable
d) comments are executed by the compiler

129. What type of comments does c++ support?
a) single line
b) multiline
c) single line and multi-line
d) reusable line

130. What is used to write multi line comment in c++?
a) /* …. */
b) /$ …. $/
c) //
d) /$ …. */

131. What is the use of the indentation in c++?
a) distinguishes between comments and code
b) r distinguishes between comments and outer data
c) distinguishes between comments and outer data
d) r distinguishes between comments and inner data

132. What will happen when we use void in argument passing?
a) It will not return value to its caller
b) It will return value to its caller
c) May or may not depend on the declared return type of the function, the passed arguments are different than the function return type
d) It will return value

133. Where does the execution of the program starts?
a) user-defined function
b) main function
c) void function
d) else function

134. What are mandatory parts in the function declaration?
a) return type, function name
b) return type, function name, parameters
c) parameters, function name
d) parameters, variables

135. Which is more effective while calling the functions?
a) call by value
b) call by reference
c) call by pointer
d) call by object

136. What is the scope of the variable declared in the user defined function?
a) whole program
b) only inside the {} block
c) the main function
d) header section

137. How many minimum number of functions should be present in a C++ program for its execution?
a) 0
b) 1
c) 2
d) 3

138. Which of the following is the default return value of functions in C++?
a) int
b) char
c) float
d) void

139. What happens to a function defined inside a class without any complex operations (like looping, a large number of lines, etc)?
a) It becomes a virtual function of the class
b) It becomes a default calling function of the class
c) It becomes an inline function of the class
d) The program gives an error

140. What is an inline function?
a) A function that is expanded at each call during execution

b) A function that is called during compile time
c) A function that is not checked for syntax errors
d) A function that is not checked for semantic analysis

141. An inline function is expanded during _____
a) compile-time
b) run-time
c) never expanded
d) end of the program

142. In which of the following cases inline functions may not word?
i) If the function has static variables.
ii) If the function has global and register variables.
iii) If the function contains loops
iv) If the function is recursive
a) i, iv
b) iii, iv
c) ii, iii, iv
d) i, iii, iv

143. When we define the default values for a function?
a) When a function is defined
b) When a function is declared
c) When the scope of the function is over
d) When a function is called

144. Where should default parameters appear in a function prototype?
a) To the rightmost side of the parameter list
b) To the leftmost side of the parameter list
c) Anywhere inside the parameter list
d) Middle of the parameter list

145. If an argument from the parameter list of a function is defined constant then _____
a) It can be modified inside the function
b) It cannot be modified inside the function
c) Error occurs
d) Segmentation fault

146. Which of the following feature is used in function overloading and function with default argument?
a) Encapsulation
b) Polymorphism
c) Abstraction
d) Modularity

147. What will be the output of the following C++ code?

```
#include<iostream>
using namespace std;

class Test
{
```

```
  protected:
    int x;
  public:
    Test (int i):x(i) { }
    void fun() const  { cout << "fun() const " << endl; }
    void fun()        {  cout << "fun() " << endl;     }
};

int main()
{
    Test t1 (10);
    const Test t2 (20);
    t1.fun();
    t2.fun();
    return 0;
}
```

a) fun()
   fun() const
b) fun() const
   fun()
c) fun()
   fun()
d) fun() const
   fun() const

148. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;

int fun(int=0, int = 0);

int main()
{
  cout << fun(5);
  return 0;
}
int fun(int x, int y) { return (x+y); }
```

a) -5
b) 0
c) 10
d) 5

149. From which function the execution of a C++ program starts?
a) start() function
b) main() function
c) new() function
d) end() function

150. Which of the following is important in a function?
a) Return type

b) Function name
c) Both return type and function name
d) The return type, function name and parameter list

151. How many ways of passing a parameter are there in c++?
a) 1
b) 2
c) 3
d) 4

152. Which is used to keep the call by reference value as intact?
a) static
b) const
c) absolute
d) virtual

153. By default how the value are passed in c++?
a) call by value
b) call by reference
c) call by pointer
d) call by object

154. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
void copy (int& a, int& b, int& c)
{
    a *= 2;
    b *= 2;
    c *= 2;
}
int main ()
 {
    int x = 1, y = 3, z = 7;
    copy (x, y, z);
    cout << "x =" << x << ", y =" << y << ", z =" << z;
    return 0;
 }
```

a) 2 5 10
b) 2 4 5
c) 2 6 14
d) 2 4 9

155. What will be the new value of x in the following C++ code?

```
#include <iostream>
using namespace std;
void fun(int &x)
 {
x = 20;
}
int main()
```

```
{
    int x = 10;
     fun(x);
     cout << "New value of x is " << x;
     return 0;
 }
```

a) 10
b) 20
c) 15
d) 36

156. What will happen when we use void in argument passing?
a) It will not return value to its caller
b) It will return value to its caller
c) Maybe or may not be return any value to its caller
d) It will return value with help of object

157. How many types of returning values are present in c++?
a) 1
b) 2
c) 3
d) 4

158. What will you use if you are not intended to get a return value?
a) static
b) const
c) volatile
d) void

159. Where does the return statement returns the execution of the program?
a) main function
b) caller function
c) same function
d) block function

160. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int max(int a, int b )
{
   return ( a > b ? a : b );
}
int main()
{
   int i = 5;
    int j = 7;
    cout << max(i, j );
    return 0;
 }
```

a) 5
b) 7

c) either 5 or 7
d) 13

161. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int mult (int x, int y)
{
    int result;
    result = 0;
    while (y != 0)
    {
        result = result + x;
         y = y - 1;
     }
     return(result);
 }
 int main ()
 {
     int x = 5, y = 5;
     cout  << mult(x, y) ;
     return(0);
 }
```

a) 20
b) 25
c) 30
d) 35

162. When will we use the function overloading?
a) same function name but different number of arguments
b) different function name but same number of arguments
c) same function name but same number of arguments
d) different function name but different number of arguments

163. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
int gcd (int a, int b)
{
    int temp;
    while (b != 0)
    {
        temp = a % b;
        a = b;
         b = temp;
     }
     return(a);
 }
 int main ()
 {
     int x = 15, y = 25;
     cout << gcd(x, y);
```

```
    return(0);
}
```

a) 15
b) 25
c) 375
d) 5

164. Which of the following permits function overloading on c++?
a) type
b) number of arguments
c) type & number of arguments
d) number of objects

165. In which of the following we cannot overload the function?
a) return function
b) caller
c) called function
d) main function

166. Function overloading is also similar to which of the following?
a) operator overloading
b) constructor overloading
c) destructor overloading
d) function overloading

167. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
void print(int i)
{
   cout << i;
}
void print(double  f)
{
   cout << f;
 }
 int main(void)
 {
print(5);
     print(500.263);
     return 0;
 }
```

**a)** 5500.263
b) 500.2635
c) 500.263
d) 500.266

168. Overloaded functions are _____
a) Very long functions that can hardly run
b) One function containing another one or more functions inside it
c) Two or more functions with the same name but different number of parameters or type
d) Very long functions

169. What will happen while using pass by reference?
a) The values of those variables are passed to the function so that it can manipulate them
b) The location of variable in memory is passed to the function so that it can use the same memory area for its processing
c) The function declaration should contain ampersand (& in its type declaration)
d) The function declaration should contain $

170. What should be passed in parameters when function does not require any parameters?
a) void
b) blank space
c) both void & blank space
d) tab space

171. What are the advantages of passing arguments by reference?
a) Changes to parameter values within the function also affect the original arguments
b) There is need to copy parameter values (i.e. less memory used)
c) There is no need to call constructors for parameters (i.e. faster)
d) All of the mentioned

172. If the user did not supply the value, what value will it take?
a) default value
b) rise an error
c) both default value & rise an error
d) error

173. Where can the default parameter be placed by the user?
a) leftmost
b) rightmost
c) both leftmost & rightmost
d) topmost

174. Which value will it take when both user and default values are given?
a) user value
b) default value
c) custom value
d) defined value

175. What will be the output of the following C++ code?

```
#include <iostream>
using namespace std;
void func(int a, bool flag = true)
{
    if (flag == true )
    {
        cout << "Flag is true. a = " << a;
    }
    else
     {
         cout << "Flag is false. a = " << a;
     }
 }
```

```
int main()
{
    func(200, false);
    return 0;
}
```

a) Flag is true. a = 200
b) Flag is false. a = 100
c) Flag is false. a = 200
d) Flag is true. a = 100

176. What we can't place followed by the non-default arguments?
a) trailing arguments
b) default arguments
c) both trailing & default arguments
d) leading arguments

177. If we start our function call with default arguments means, what will be proceeding arguments?
a) user argument
b) empty arguments
c) default arguments
d) user & empty arguments

178. What is the default return type of a function?
a) int
b) void
c) float
d) char

179. What is the maximum number of arguments or parameters that can be present in one function call?
a) 64
b) 256
c) 255
d) 16

180. To which does the function pointer point to?
a) variable
b) constants
c) function
d) absolute variables

181. What will we not do with function pointers?
a) allocation of memory
b) deallocation of memory
c) both allocation & deallocation of memory
d) finds memory status

182. What is the mandatory part to present in function pointers?
a) &
b) return values
c) data types
d) $

183. which of the following can be passed in function pointers?
a) variables
b) data types
c) functions
d) objects

184. What is the meaning of the following declaration?

```
int(*ptr[5])();
```

a) ptr is pointer to function
b) ptr is array of pointer to function
c) ptr is pointer to such function which return type is array
d) ptr is pointer to array of function

185. How many types of access specifiers are present in C++?
a) 3
b) 4
c) 1
d)5

## Unit I - Principles of OOP & Introduction

1. Which is not a feature of OOP in general definitions?
   a) Efficient Code
   b) Code reusability
   c) Modularity
   d) Duplicate/Redundant data
   Ans: d
2. Which feature of OOP indicates code reusability?
   a) Abstraction
   b) Polymorphism
   c) Encapsulation
   d) Inheritance
   Ans: D
3. Which of the following is not true about polymorphism?
   a) Helps in redefining the same functionality
   b) Increases overhead of function definition always
   c) It is feature of OOP
   d) Ease in readability of program
   Ans: B
4. Which among the following doesn't come under OOP concept?
   a) Data hiding
   b) Message passing
   c) Platform independent
   d) Data binding
   Ans: C

5. Which is the correct syntax of inheritance?
   a) class base_classname :access derived_classname{ /*define class body*/ };
   b) class derived_classname : access base_classname{ /*define class body*/ };
   c) class derived_classname : base_classname{ /*define class body*/ };
   d) class base_classname : derived_classname{ /*define class body*/ };
   Ans: B
6. The feature by which one object can interact with another object is _____
   a) Message reading
   b) Message Passing
   c) Data transfer
   d) Data Binding

   Ans: B

7. Which of the following is not true about polymorphism?
   a) Helps in redefining the same functionality
   b) Increases overhead of function definition always
   c) It is feature of OOP
   d) Ease in readability of program
   Ans: B
8. What is encapsulation in OOP?
   a) It is a way of combining various data members and member functions that operate on those data members into a single unit
   b) It is a way of combining various data members and member functions into a single unit which can operate on any data
   c) It is a way of combining various data members into a single unit
   d) It is a way of combining various member functions into a single unit
   Ans: A
9. What is an abstraction in object-oriented programming?
   a) Hiding the implementation and showing only the features
   b) Hiding the important data
   c) Hiding the implementation
   d) Showing the important data
   Ans: A
10. Which among the following can show polymorphism?
    a) Overloading &&
    b) Overloading <<
    c) Overloading ||
    d) Overloading +=
    Ans: B
11. In which access should a constructor be defined, so that object of the class can be created in any function?
    a) Any access specifier will work
    b) Private
    c) Public
    d) Protected
    Ans: C
12. The C++ keyword for declaring a variable that contains a decimal point is _____
    a. dec
    b. decimal
    c. float
    d. floater

Ans: C

13. When accessing a structure member, the identifier to the left of the dot operator is the name of
    a. A structure member
    b. A structure tag
    c. A structure variable
    d. The keyword structure
    Ans: C

14. Programmers prefer to declare almost all variables _____
    a. At the beginning of each function
    b. Globally
    c. On one line
    d. With cryptic names
    e. Ans: A
15. _____ refers to the process of locating and removing the errors in a program
    a. Analyzing
    b. Correcting
    c. Debugging
    d. Executing
    e. Ans: C
16. A pointer is
    a. The address of a variable
    b. An indication of the variable to be accessed next
    c. A variable for storing addresses
    d. The data type of an address variable
    e. Ans: C
17. The feature that allows the same operations to be carried out differently depending on the object is _____
    a. Polymorphism
    b. Polygamy
    c. Inheritane
    d. Multitasking
    e. Ans: A
18. The most efficient data type for a variable that stores the letter C is the _____ data type
    a. Character
    b. Double
    c. Float
    d. Long Integer
    e. Ans: A
19. Which feature of OOP reduces the use of nested classes?
    a) Inheritance
    b) Binding
    c) Abstraction
    d) Encapsulation
    Ans: A
20. Which of the following correctly declares an array in C++?
    a) array{10};
    b) array array[10];
    c) int array;
    d) int array[10];
    Ans: D
21. How are the constants declared?
    a) const keyword

b) #define preprocessor
c) both const keyword and #define preprocessor
d) $define
Ans: C

22. Which of the following gives the memory address of the first element in array?
a) array[0];
b) array[1];
c) array(2);
d) array;
Ans: D

23. Which of the following accesses the seventh element stored in array?
a) array[6];
b) array[7];
c) array(7);
d) array;
Ans: A

24. The operator used for dereferencing or indirection is ____
a) *
b) &
c) ->
d) –>>
Ans: A

25. Which of the following is illegal?
a) int *ip;
b) string s, *sp = 0;
c) int i; double* dp = &i;
d) int *pi = 0;
Ans: C

26. Identify the type of variables.

```
typedef char* CHAR;
  CHAR p,q;
```

a) char*
b) char
c) CHAR
d) unknown
Ans: A

27. Which of the following numerical value(s) is(are) invalid constant(s)?
a. 0.7
b. 9.3el2
c. 27,512
d. 12345678
Ans: C

28. The bitwise OR operator is a
a. Unary operator
b. Binary operator
c. Ternary operator
d. Octal operator
e. Ans: B

29. The comma operator (,) is primarily used in conjunction with
a. 'for' statement
b. 'if-else' statement
c. 'do-while' statement

d. None of the above
e. Ans: A

30. The C++ _____ function generates random numbers
    a. Generate()
    b. genRand
    c. rand
    d. srand
    e. Ans: D
31. To use the strcpy function, you must include the _____ header file in your program
    a. Assign.h
    b. Copy.h
    c. String.h
    d. Strcopy.h
    e. Ans: C

Unit II – Classes & Objects

32. The functions go in the _____ section of a class definition
    a. Declaration
    b. Implementation
    c. Prototype
    d. Functioning
    e. Ans: B
33. If you want to use a class to define objects in many different programs, you should define the class in a C++ _____ file.
    a. Header
    b. Program
    c. Source
    d. Text
    e. Ans: A
34. Which access specifier is usually used for data members of a class?
    a) Protected
    b) Private
    c) Public
    d) Default
    Ans: B
35. How to access data members of a class?
    a) Dot, arrow or direct call
    b) Dot operator
    c) Arrow operator
    d) Dot or arrow as required
    Ans: D
36. Which operator can be used to free the memory allocated for an object in C++?
    a) Unallocate
    b) Free()
    c) Collect
    d) delete
    Ans: D
37. Which of the following is not a property of an object?
    a) Properties

b) Names
c) Identity
d) Attributes
Ans: B

38. Which is correct syntax for declaring pointer to object?
    a) *className objectName;
    b) className* objectName;
    c) className objectName();
    d) className objectName;
    Ans: B
39. When is the memory allocated for an object?
    a) At declaration of object
    b) At compile time
    c) When object constructor is called
    d) When object is initialized to another object
    Ans: C
40. Which of the following function can be used for dynamic memory allocation of objects?
    a) malloc()
    b) calloc()
    c) create()
    d) both malloc() and calloc()
    Ans: D
41. Which operator can be used to check the size of an object?
    a) sizeof(objectName)
    b) size(objectName)
    c) sizeofobject(objectName)
    d) sizedobject(objectName)
    Ans: A
42. Which operator is used in pointer to member function?
    a) .*
    b) ->*
    c) Both .* & ->*
    d) $*
    Ans:C
43. To hide a data member from the program, you must declare the data member in the _____ section of the class
    a. Concealed
    b. Confidential
    c. Hidden
    d. Private
    f. Ans: D
44. Private data can be accessed by
    a. Class member functions
    b. Functions in derived classes
    c. Both (a) and (b)
    d. Neither (a) nor (b)
    e. Ans: A
45. Which is used to define the member of a class externally?
    a) :
    b) ::
    c) #
    d) !!$

Ans: B

46. Which of the following is a valid class declaration?
    a) class A { int x; };
    b) class B { }
    c) public class A { }
    d) object A { int x; };
    Ans: A
47. The data members and functions of a class in C++ are by default _____
    a) protected
    b) private
    c) public
    d) public & protected
    Ans: B
48. Which category of data type a class belongs to?
    a) Fundamental data type
    b) Derived data type
    c) User defined derived data type
    d) Atomic data type
    Ans: C

Unit III – Constructors & Op Overloading

49. The most common operation used in constructors is
    a. Addition
    b. Overloading
    c. Assignment
    d. Polymorphism

    Ans: C

50. A constructor initialization list produces similar results to
    a. Overriding
    b. Assignment
    c. Redeclaring
    d. Output

    Ans: B

51. A function that is called automatically each time an object is created is a(n)
    a. Constructor
    b. Contractor
    c. Builder
    d. Architect
    e. Ans: A
52. In which access should a constructor be defined, so that object of the class can be created in any function?
    a) Any access specifier will work
    b) Private
    c) Public
    d) Protected
    Ans: C

53. The copy constructors can be used to _____
    a) Copy an object so that it can be passed to another primitive type variable
    b) Copy an object for type casting
    c) Copy an object so that it can be passed to a function
    d) Copy an object so that it can be passed to a class
    Ans: C
54. What happens when an object is passed by reference?
    a) Destructor is called at end of function
    b) Destructor is called when called explicitly
    c) Destructor is not called
    d) Destructor is called when function is out of scope
    Ans: C
55. Object being passed to a copy constructor _____
    a) Must not be mentioned in parameter list
    b) Must be passed with integer type
    c) Must be passed by value
    d) Must be passed by reference
    Ans: D
56. Which among the following is not a necessary condition for constructors?
    a) Its name must be same as that of class
    b) It must not have any return type
    c) It must contain a definition body
    d) It can contains arguments
    Ans: C
57. Which feature in OOP is used to allocate additional functions to a predefined operator in any language?
    a) Function Overloading
    b) Function Overriding
    c) Operator Overloading
    d) Operator Overriding
    Ans: C
58. What is the syntax to overload an operator?
    a) className::operator<operatorSymbol>(parameters)
    b) className:operator<operatorSymbol>(parameters)
    c) className.operator<operatorSymbol>(paramteres)
    d) className->operator<operatorSymbol>(parameters)
    Ans: A
59. Which operator among the following can be overloading using only member function?
    a) Assignment operator
    b) Addition operator
    c) Subtraction operator
    d) Multiplication and division operator
    Ans: A
60. Which operator among the following can be overloaded using both friend function and member function?
    a) Assignment operator
    b) Subscript
    c) Member selection (arrow operator)
    d) Modulus operator
    Ans: D

61. Which of the following constructors are provided by the C++ compiler if not defined in a class?
    a) Copy constructor
    b) Default constructor
    c) Assignment constructor
    d) All of the mentioned
    Ans: D
62. A function that is called automatically each time an object is destroyed is a
    a. Constructor
    b. Destructor
    c. Destroyer
    d. Terminator
    f. Ans: B
63. What is the order of Constructors call when the object of derived class B is declared, provided class B is derived from class A?
    a) Constructor of A followed by B
    b) Constructor of B followed by A
    c) Constructor of A only
    d) Constructor of B only
    Ans: A

Unit IV – Inheritance

64. Inheritance is the principle that
    a. Classes with the same name must be derived from one another
    b. Knowledge of a general category can be applied to more specific objects
    c. C++ functions may be used only if they have logical predecessors
    d. One function name may invoke different methods
    e. Ans: B
65. In multilevel inheritance, which is the most significant feature of OOP used?
    a) Code efficiency
    b) Code readability
    c) Flexibility
    d) Code reusability
    Ans: D
66. Which type of members can't be accessed in derived classes of a base class?
    a) All can be accessed
    b) Protected
    c) Private
    d) Public
    Ans: C
67. Which among the following best describes the Inheritance?
    a) Using the data and functions into derived segment
    b) Using already defined functions in a programming language
    c) Using the code already written once
    d) Copying the code already written
    Ans: A
68. Single level inheritance supports _____ inheritance.
    a) Language independency

b) Multiple inheritance
c) Compile time
d) Runtime
Ans: D

69. If in multiple inheritance, class C inherits class B, and Class B inherits class A. In which sequence are their destructors called if an object of class C was declared?
a) ~A() then ~B() then ~C()
b) ~C() then ~A() then ~B()
c) ~C() then ~B() then ~A()
d) ~B() then ~C() then ~A()
Ans: C

70. A derived class may also be called a
    a. Subclass
    b. Super class
    c. Parent class
    d. Base class
    e. Ans: A

71. What is meant by multiple inheritance?
a) Deriving a base class from derived class
b) Deriving a derived class from base class
c) Deriving a derived class from more than one base class
d) Deriving a derived base class
Ans: C

72. Which symbol is used to create multiple inheritances?
a) Dot
b) Comma
c) Dollar
d) star
Ans: B

73. What are the things are inherited from the base class?
a) Constructor and its destructor
b) Operator=() members
c) Friends
d) All of the mentioned
Ans: D

74. If a class is derived privately from a base class then _____
a) no members of the base class is inherited
b) all members are accessible by the derived class
c) all the members are inherited by the class but are hidden and cannot be accessible
d) no derivation of the class gives an error
Ans: C

75. Which is the correct syntax of declaring a virtual function?
a) virtual int func();
b) virtual int func(){};
c) inline virtual func();
d) inline virtual func(){};
Ans: A

76. What is the order of Destructors call when the object of derived class B is declared, provided class B is derived from class A?
a) Destructor of A followed by B
b) Destructor of B followed by A

c) Destructor of A only
d) Destructor of B only
Ans: B

77. When a child class inherits traits from more than one parent class, this type of inheritance is called _____ inheritance.
    a. Hierarchical
    b. b. Hybrid
    c. c. Multilevel
    d. d. Multiple
    f. Ans: D
78. What is the difference between protected and private access specifiers in inheritance?
    a. private member is not inheritable and not accessible in derived class.
    b. protected member is inheritable and also accessible in derived class.
    c. Both are inheritable but private is accessible in the derived class.
    d. Both are inheritable but protected is not accessible in the derived class.
    e. Ans: B
79. When a base class is privately inherited by the derived class, then_____ .
    a. protected members of the base class become private members of derived class
    b. b. public members of the base class become private members of derived class
    c. c. both a and b
    d. d. only b
    e. Ans: C
80. class X, class Y and class Z are derived from class BASE. This is _____ inheritance.
    a. Multiple
    b. b. Multilevel
    c. c. Hierarchical
    d. d. Single
    e. Ans: C
81. When base class is derived in protected mode, then_____ .
    1 public members of base class become protected members of derived class.
    2. public members of base class become public members of derived class.
    3. protected members of base class become protected members of derived class.
    a. 2
    b. 3
    c. 1 and 2 only
    d. 1 and 3 only
    e. Ans: D
82. During a class inheritance in CPP, if the visibility mode or mode of derivation is not provided, then by default visibility mode is _____.

a. Public
b. Protected
c. Private
d. Friend
e. Ans: C

83. Private members of the class are not inheritable.
    a. True
    b. False
    c. Ans: B
84. Can we pass parameters to base class constructor though derived class or derived class constructor?
    a. Yes
    b. No

g. Ans: A

85. Can struct be used as Base class for inheritance ?
    a. Yes
    b. No

    Ans: A

86. In case of inheritance where both base and derived class are having constructor and destructor, then which if the following are true ?

    1. Constructors are executed in their order of derivation
    2. Constructors are executed in reverse order of derivation
    3. Destructors are executed in reverse order of derivation
        a. 2
        b. 3
        c. & (2)
        d. & (3)

    Ans: D

Unit V – Functions

87. Assume a program contains a void function named displayName, which requires no formal parameters. Which of the following is a correct function prototype for this function?
    a. displayName;
    b. displayName(void);
    c. void displayName;
    d. void displayName();
    Ans: D

88. If you assign a default value to any variable in a function prototype's parameter list, then _____
    e. All other parameters in the function prototype must have default values
    f. All parameters to the right of that variable must have default values
    g. All parameters to the left of that variable must have default value
    h. No other parameters in that prototype can have default values
    i. Ans: B

89. Which keyword is used to declare virtual functions?
    a) virt
    b) virtually
    c) virtual
    d) anonymous
    Ans: C

90. What happens if non static members are used in static member function?
    a) Executes fine
    b) Compile time error
    c) Executes if that member function is not used
    d) Runtime error
    Ans: B

91. What is friend member functions in C++?
    a) Non-member functions which have access to all the members (including private) of a class
    b) Member function which doesn’t have access to private members

c) Member function which can modify any data of a class
d) Member function which can access all the members of a class
Ans: A

92. Which feature of OOP is exhibited by the function overriding?
a) Polymorphism
b) Encapsulation
c) Abstraction
d) Inheritance
Ans: A

93. How to access the private member function of a class?
a) Using class address
b) Using object of class
c) Using object pointer
d) Using address of member function
Ans: D

94. _____ underlines the feature of Polymorphism in a class.
a) Virtual Function
b) Inline function
c) Enclosing class
d) Nested class
Ans: A

95. Where the virtual function should be defined?
a) Twice in base class
b) Derived class
c) Base class and derived class
d) Base class
Ans: D

96. In which access specifier should a virtual function be defined?
a) Private
b) Public
c) Protected
d) Default
Ans: B

97. he resolving of virtual functions is done at _____
a) Compile time
b) Interpret time
c) Runtime
d) Writing source code
Ans: C

98. Virtual functions can never be made _____
a) Static function
b) Parameterized function
c) Default argument function
d) Zero parameter function
Ans: A

99. The virtual functions must be declared and defined in _____ class and overridden in _____ class.
a) Base, base
b) Derived, derived
c) Derived, base
d) Base, derived

Ans: D

100. Simple member functions are _____
a) Ones defined simply without any type
b) Ones defined with keyword simple
c) Ones that are implicitly provided
d) Ones which are defined in all the classes
Ans: A

101. How can static member function can be accessed directly in main() function?
a) Dot operator
b) Colon
c) Scope resolution operator
d) Arrow operator
Ans: C

102. Which among the following best describes the inline member functions?
a) Functions defined inside the class only
b) Functions with keyword inline only
c) Functions defined outside the class
d) Functions defined inside the class or with the keyword inline
Ans: D

103. What are friend member functions (C++)?
a) Member function which can access all the members of a class
b) Member function which can modify any data of a class
c) Member function which doesn't have access to private members
d) Non-member functions which have access to all the members (including private) of a class
Ans: D

104. Which among the following is true?
a) Member functions can never be private
b) Member functions can never be protected
c) Member functions can never be public
d) Member functions can be defined in any access specifier
Ans: D

105. Which keyword is used to define the inline member function?
a) no keyword required
b) inline
c) inlined
d) line
Ans: B

106. Inline functions are avoided when _____
a) function contains static variables
b) function have recursive calls
c) function have loops
d) all of the mentioned
Ans: D

107. What are Templates in C++?
a) A feature that allows the programmer to write generic programs
b) A feature that allows the programmer to write specific codes for a problem
c) A feature that allows the programmer to make program modular
d) A feature that does not add any power to the language
Ans:A

108. What is a virtual function in C++?

a) Any member function of a class

b) All functions that are derived from the base class

c) All the members that are accessing base class data members

d) All the functions which are declared in the base class and is re-defined/overridden by the derived class

Ans: D

109. What are mandatory parts in the function declaration?

a) return type, function name

b) return type, function name, parameters

c) parameters, function name

d) parameters, variables

Ans: A

110. An inline function is expanded during _____

a) compile-time

b) run-time

c) never expanded

d) end of the program

Ans: A

111. Where should default parameters appear in a function prototype?

a) To the rightmost side of the parameter list

b) To the leftmost side of the parameter list

c) Anywhere inside the parameter list

d) Middle of the parameter list

Ans: A

112. If an argument from the parameter list of a function is defined constant then _____

a) It can be modified inside the function

b) It cannot be modified inside the function

c) Error occurs

d) Segmentation fault

Ans: B

Section B

Unit 1

1. What is Object Oriented Programming? Give any two differences between object oriented programming and procedural programming.
2. List the advantages of OOP.
3. Write a note on application and benefits of OOP.
4. Explain the features of OOP
5. Explain basic data types supported by C++.
6. Explain increment and decrement operators with example
7. Explain for loop structure with example
8. Explain the use of break and continue statements in C++
9. Explain applications and advantages of OOP
10. Explain classification of data types in C++
11. Explain arithmetic, relational and logical operators in C++
12. Explain various bitwise operators available in C++
13. Explain any loop control structures with syntax and example.
14. Write the difference between C language and C++.
15. What is OOP? How is it different from the procedure-oriented programming.
16. Write a program to find the factorial of a number?
17. Add the correct data type for the following variables.
    i. _____ myNum = 9;
    ii. _____myDoubleNum = 8.99;
    iii. _____myLetter = 'A';
    iv. _____myBool = false;
    v. _____myText = "Hello World";
18. What are the general rules for naming variables?
19. Write a C++ program to check whether number is even or odd.
20. Write a C++ program to generate multiplication of 10.
21. What are the advantages of C++?
22. What are the different data types present in C++?
23. What is the C++ OOPs concept?
24. How many types of modifiers are there in C++? Explain.
25. Write a short note of History of C++
26. Explain C++ structures with example.
27. What is variables in C++ explain its rules
28. What are the different types of modifiers in C++
29. Explain input/output statements with example
30. What are the advantages of function prototypes in C++
31. Explain parameter passing in C++
32. Explain call by reference in C++
33. What is the difference between while loop and do while loop?

## Unit 2

1. Explain C++ Classes and objects.

2. What are the difference between classes and objects in C++?
3. Write a short note on Classes and objects in C++
4. What is a class? How do you specify a class?
5. What is a class? How do you declare a public class in C++?
6. What is member function and its types?
7. What is the difference between function and member function in C++?
8. Explain static member function in C++ with example.
9. What are the advantages of arrays?
10. Explain array and its types with example.
11. Illustrate structures with an example
12. Write a program to accept n numbers, store it in an array and display it.
13. Explain how to pass arrays to functions with a suitable example.
14. What is friend function? What are the merits and demerits of using friend| function
15. Write a program using friend functions to add, subtract and multiply two complex numbers

Unit 3

1. What is the difference between constructors and de-constructors in C++
2. What are constructors in C++?
3. What are de-constructors in C++?
4. What is Parameterized constructors?
5. What is Parametric constructors? Explain with examples.
6. What is the difference between default and parameterized constructors?
7. What are constructors and destructors in C++?.Give suitable example.
8. What are the uses of constructors and destructors?
9. What is the difference between operator functions and normal functions?
10. What are the rules of operator overloading in C++?
11. Explain the difference between function overloading and operator overloading in C++.
12. What is operator overloading?Give a suitable example.
13. What is function overloading? Explain with example.
14. What is the difference between unary and binary operators?
15. What are unary operators?
16. What are binary operators?
17. What are the characteristic of friend function in C++?
18. What is a friend function in C++?
19. What are the advantages of friend function in C++?
20. How do you declare a friemd function in C++?
21. What are the restrictions on operator overloading?
22. What are the rules for operator overloading in C++?
23. What are the characteristics of constructor?
24. What are the rules for overloading an operator?
25. How do you overload a unary operator using friend function? Explain with example
26. Define a class String. Using overloaded operator = = check whether two strings are equal or not .
27. Explain how multiple constructors are defined in a class with example.
28. What is meant by constructor overloading? Explain with code example.
29. Write a program to generate fibonacci numbers by overloading ++ operator
30. How do you overload a binary operator using friend function ? Explain with example.

31. Explain basic to class type conversion with an example.
32. Explain class to basic type conversion with an example.
33. Explain one class another class conversion with an example

Unit 4

1. What are the different types of inheritance?
2. What is inheritance in C++? Explain with example.
3. What are the advantages of inheritance in C++?
4. What is the difference between base class and derived class in C++?
5. Explain multi-level inheritance with example.
6. Explain Hybrid inheritance with example.
7. Explain Hierarchical inheritance with example.
8. Explain single inheritance with example
9. Explain multiple inheritance with example
10. Explain abstract class with example.
11. Explain visibility of private , protected and public members in different modes of inheritance

Unit 5

1. What is virtual function in C++?
2. What are the rules of virtual function in C++?
3. What are pure virtual functions?. Give a suitable example.
4. What are the different types of templates in C++? Explain
5. Explain function template with example
6. Explain class template with example.
7. What are the advantages of using templates?
8. What are the disadvantages of using templates?
9. How can we inherit template class c++?
10. Explain the concept of overloading a function in C++ with example

10 marks

1. Explain the different types of loop control structures with syntax and example.

2. What is inheritance in C++ and name the different types of inheritance?

3. What is a constructor? Explain different types of Constructors?

4. Write a program to demonstrate friend function in C++.

5. What is role of manipulators in C++. Write down different manipulators in C++.

5. Differentiate between formatted and unformatted I/O. Discuss its different functions.

6.Differentiate between nested if-else and switch statement.

7.Compare and contrast for, whileand do-while looping statements.

8. Differentiate between break and continue statement.

9. Differentiate between nested if-else and switch statement.

10. Compare and contrast for, whileand do-while looping statements.

11. Differentiate between break and continue statement.

12. Explain the concept of polymorphism by an example in C++.

13. Write a C++ program to sort integer and float array elements in ascending order by using function overloading.

14. Design a base class Person (name, address) and derived class as Student(rollno, percentage). Write member functions to accept and display information of student.(use virtual function).

15. ) Define a C++ class Cstring to represent a string. Define parameterized constructor and member function to display string . Overload unary to change the case of the string object.

16. Design a class Student. Include data members rollno, name, city and age. Write member functions:

i) To accept information of ‘n’ students

ii) To display information of ‘n’ students

iii) To search details of a student using rollno

(use array of objects)

17. Explain unary and binary operator overloading with the help of member functions.

18. Distinguish between the following terms:

(a) Object and classes

(b) Data abstraction and data encapsulation

(c) Inheritance and polymorphism

(d) Dynamic binding and message passing

19. What is friend function? How it is different from member function? Explain with an example.

20. What are virtual functions and pure virtual functions? Explain the use ofv having abstract classes.