



**INDIAN SCHOOL SOHAR**  
**UNIT TEST**  
**COMPUTER SCIENCE**

**CLASS: XII**

**DATE: 19/05/2019**

**MAX. MARKS: 50**

**DURATION: 2 HOURS**

1. Write the type of C++ tokens (keywords or user defined identifiers) from the following: 1  
(i) case (ii) \_delete (iii) WHILE (iv) typedef
2. What are inline functions? Give appropriate example. 1
3. What is a destructor? Why is it needed? 1
4. Explain scope resolution operator with appropriate example. 1
5. Explain the following file mode constants: ios::nocreate and ios:: binary. 2
6. Explain setw() manipulator with appropriate example. 2
7. What is the role of parameter/argument passed in a function? Can a default value be assigned to a parameter? If yes, then justify the answer with a suitable example. 2
8. What do you understand by method/function overloading or functional polymorphism? Explain with appropriate example. 2
9. What are the advantages of Object Oriented Programming? Write any 4 advantages. 2
10. What do you mean by Inheritance? Explain different types of inheritance briefly. 4
11. What is the difference between an object and a class? Explain with appropriate example. 2
12. Differentiate between global variables and local variables with appropriate example. 2
13. Write a C++ program that uses an area() function for the calculation of area of a triangle or a rectangle or a square using the concept of polymorphism. Number of sides (3 for triangle, 2 for rectangle and 1 for square) suggest about the shape for which area is to be calculated. 4  
**(NOTE:  $s = \frac{\text{side1} + \text{side2} + \text{side3}}{2}$ . Area of Triangle =  $\sqrt{s(s-a)(s-b)(s-c)}$ , Area of Rectangle =  $\text{side1} * \text{side2}$ , Area of Square =  $\text{side1} * \text{side1}$ )**
14. Observe the following C++ code and answer the questions (i) & (ii). Assume all necessary files are included: 2  
class FICTION  
{  
    long FCode;  
    char FTitle[20];  
    float FPrice;  
    **public:**  
    FICTION() //Member Function 1  
    { cout << "Bought" << endl;  
      FCode = 100; strcpy(FTitle, "Alone!"); FPrice=500.00; }  
    FICTION(int c, char T[], float P) //Member Function 2  
    {  
      FCode = c;      strcpy(FTitle, T);      FPrice = P;  
    }  
    void Increase(float P)  
    { FPrice += P; }  
    void show()  
    { cout << FCode << ":" << FTitle << ":" << FPrice << endl; }  
    ~Fiction()  
    { cout << "Fiction removed!" << endl; }  
};  
void main()

```

{
    FICTION F1, F2(105, "EMPTY!", 625.00);
    for(int i = 0; i < 4; i++)
    {
        F1.Increase(20); F2.Increase(35);
        F1.show(); F2.show();    }
}

```

- (i) Which specific concept of object oriented programming out of the following is illustrated by Member Function 1 and Member Function 2 combined together?
- (ii) How many times the message "Fiction removed!" will be displayed after executing the above C++ code?

15. Answer the questions from (i) to (iv) based on the given below code (assume all necessary header files are included in the program:

4

```

class Place
{
    int place_id;
    char place_name[30];
    protected:
    int pl_pop;
    public:
    Place();
    void get_pop();
    void new_place();
    void show_place();
};
class State : public Place
{
    int st_id;
    char st_name[25];
    protected:
    int st_pop;
    public:
    State();
    void new_state();
    void show_state();
};
class Country : private State
{
    int cntry_id;
    char cntry_name[25];
    public:
    Country();
    void new_cntry();
    void disp_cntry();
};

```

- i. Write name of the class whose constructor is invoked first on the creation of new object of class Country.
- ii. Write name of the data members which are accessible through the object of class Country.
- iii. List name of the members which are accessible through the member function "void new\_cntry()".

iv. What will be the size (in bytes) of an object of class Country & State respectively.

16. Declare a class and write C++ code to represent bank account of 10 customers with the following data members. Name of the depositor, Account Number, Type of account (S for savings and C for current), Balance amount.

6

The class also contains member functions to perform the following tasks:

- To initialize data members (constructor)
- To input values of data members
- To deposit money
- To withdraw money after checking the balance (minimum balance to maintain Rs. 1000)
- To display the data members.

17. Write a C++ program to search a record by “emp\_id” in an existing file “emp.dat”. Define a class employee with following specification:

6

**Private Members:**

emp\_id (unsigned short integer), emp\_nm (string of 25 characters), basic (unsigned integer), emp\_desig (string of 15 characters)

**Public Members:**

getdata() – to input employee information

putdata() – to display employee information

get\_empid() – to get the employee id of the employee

If the employee id is found then employee information should be displayed in a formatted manner. Otherwise error message “Employee with <employee id> is not found in the file!” is displayed.

18. Write the output of the following C++ programs

a. struct Tkts

2

```
{
    char level;
    int price;
};
void Find(Tkts &T)
{
    if(T.level == 'A')
        T.price += 60;
    else if(T.level == 'B')
        T.price += 45;
    else if(T.level == 'C')
        T.price += 35;
    cout << T.level << " :: " << T.price << endl;
}
void main()
{
    Tkts Tue[] = { {'C', 260}, {'A', 315}, {'B', 350} };
    for(int cnt = 0; cnt <= 2; cnt++)
        Find(Tue[cnt]);
}
```

b. class A

2

```
{
    public:
    A() { cout << "A"; }
    ~A() { cout << "~A"; }
};
```

```

class B
{
    public:
    B() { cout << "B"; }
    ~B() { cout << "~B"; }
};
class C
{
    public:
    C() { cout << "C"; }
    ~C() { cout << "~C"; }
    private:
    B c1;
    A c2;
};
class D
{ public:
    D() { cout << "D"; }
    ~D() { cout << "~D"; }
};
class E : public C
{ public:
    E(){ cout << "E"; }
    ~E() { cout << "~E"; }
    private:
    D e1;
    C e2;
};
void main()
{ E e; }
c. void chng(int *s)
{
    for(int i = 0; i < 4; i++){
        if(*s < 40){
            if(*s%2 == 0)
                *s = *s + 10;
            else
                *s = *s + 11; }
        else{
            if(*s%2 == 0)
                *s = *s - 10;
            else
                *s = *s - 11; }
        cout << "\n *s = " << *s;
        s++; } // End of for loop
}
void main()
{
    int score[] = {35, 70, 48, 63};
    chng(score); }

```

2