

[This question paper contains 8 printed pages.]

Your Roll No.....

Sr. No. of Question Paper : 1786

L

Unique Paper Code : 2342011204

Name of the Paper : Programming Using C++ (Adm. in 2024)

Name of the Course : **B.Sc. (H) Computer Science**

Semester : II

Duration : 3 Hours

Maximum Marks : 90

Instructions for Candidates

1. Write your Roll No. on the top immediately on receipt of this question paper.
2. **Section-A** is compulsory.
3. Attempt any **four** Questions from **Section B**.
3. Parts of question must be answered together.

Section A

Q1. a) What is a void pointer in C++? Explain with the help of a suitable code. (2)

b) Find the output of the following code: (3)

```
#include <iostream>
using namespace std;
int main() {
    int arr[3] = {8, 20, 30};
    int *p = arr;
    cout<<*p <<endl;
    cout<<*p+1 << "\t" << *(p+2) << "\t" <<endl;
    return 0;
}
```

c) What is an inline function? Write a C++ program using an inline function to calculate the cube of a number. (3)

d) Write a C++ function to swap the values of two variables using call by reference. (3)

e) Find the output of the following code(s): (3*1)

i.

```
#include <iostream>
using namespace std;
int main() {
    int a = 5;
    cout<< (a > 3 ? a + 2 : a - 2);
    return 0;
}
```

ii.

```
#include <iostream>
using namespace std;
int main() {
    char ch = 'A';
}
```

```

        cout<<ch + 2;
        return 0;
    }

    iii.
#include <iostream>
using namespace std;
int main() {
    int x = 2;
    cout<< x * x++ << " " << ++x;
    return 0;
}

```

- f) State whether true or false: (3*1)
- Destructors can be declared as private members of the class.
 - Constructors can have default arguments.
 - Constructors can be used to return values from the function.

- g) Find the output of the following code: (3)

```

#include <iostream>
using namespace std;
int func(int x, int y = 2, int z = 3) {
    return x * y + z;
}
int main() {
    cout<<func(4) <<endl;
    cout<<func(4, 5) <<endl;
    cout<<func(4, 5, 6) <<endl;
    return 0;
}

```

- h) Write a C++ function to find the length of a string without using any (3)
inbuilt string functions.
- i) How does the visibility modes of inheritance differ in the following (3*1)
derived classes:
- class D1: private B
 - class D2: protected B
 - class D3: public B

- j) Find the output of the following code: (2*2)

```

i.
#include <iostream>
using namespace std;
int main() {
    int x = 1, y = 2;

```

```

switch(x) {
    case 1:
        switch(y) {
            case 1: cout<< "Inner1"; break;
            case 2: cout<< "Inner2"; break;
        }
        break;
    default: cout<< "Outer";
}
return 0;}

```

ii.

```

#include <iostream>
using namespace std;
int main() {
    int i, j = 0;
    for(i = 1; i <= 3; i++) {
        j = i;
        while(j > 0) {
            cout<< j << " ";
            j--;
        }
    }
    return 0;
}

```

Section B

Q2. a) What is polymorphism? Explain early binding and late binding in the context of polymorphism. (5)

b) Find the output of the following code: (5)

```

#include <iostream>
using namespace std;
int x = 100;
class Test {
public:
    int x;
    Test() { x = 50; }
    void show() {
        int x = 10;
        cout<< x << " ";
    }
}

```

```

cout<< this->x << " ";
cout<< ::x << " ";
cout<< x + this->x + ::x << endl;
    }
};

int main() {
    Test t;
    t.show();
    return 0;
}

```

- c) Create a base class Shape with one data member radius and a virtual function area(). Derive a class Circle using public inheritance and override the function area() to calculate the area of the circle. Also, create another derived class Square with data member side and override the function area(). Demonstrate how you can call the area functions of both derived classes using a single pointer to the base class. (5)

- Q3. a) Find the output of the following code: (5)

```

#include <iostream>
using namespace std;
int main() {
    int a[] = {1, 2, 3, 4};
    int *p = a;
    cout<< *p++ << " ";
    cout<< ++*p << " ";
    cout<< *p;
    return 0;
}

```

- b) Write a C++ program to accept two integers a and b from the user and perform the division a / b. Before performing the division, appropriate conditions should be checked such that if the denominator b is zero, an integer exception is thrown, and if the numerator a is negative, a character exception is thrown. Use multiple catch blocks to handle each type of exception separately and display suitable error messages for each case. (5)
- c) Write a C++ program to initialize an integer array of size 10. Find the smallest number and print its position. (5)

- Q4. a) Write a C++ program to implement following specifications: (10)

- i. Class Distance1
 - a. Private data member: meters.
 - b. Single parameter Constructor.
 - c. Public member function, init1() to initialize the value.

- ii. Declare a function void addDistance(Distance1) in the class that:
 - a. Accepts object of Distance1.
 - b. Adds their distances.
 - c. Displays the total distance.
 - iii. In the main() function:
 - a. Create objects of class
 - b. Initialize them with values
 - c. Call the addDistance function to display the result.
- b) Write a function DiagonalMatrix() that accepts a square matrix A (5) and its order n. The function should convert matrix A into a diagonal matrix by assigning 0 to all non-diagonal elements.
- Q5. a) Write a program in C++ to print the following pattern. Take number of (5) rows as input from the user.

```
A
B C
D E F
G H I J
```

- b) Find the output of the following code: (5)

```
#include <iostream>
using namespace std;
class Base {
public:
    virtual void show() {
        cout<< "Base class show\n";
    }
};
class Derived1 : public Base {
public:
    void show() {
        cout<< "Derived1 class show\n";
    }
};
class Derived2 : public Base {
public:
    void show() {
        cout<< "Derived2 class show\n";
    }
};
int main() {
    Base *b;
    Derived1 d1;
```

```

    Derived2 d2;
    Base baseObj;
    b = &baseObj;
    b->show();
    b = &d1;
    b->show();
    b = &d2;
    b->show();
    b = &d1;
    b->show();
    b = &baseObj;
    b->show();
    return 0;
}

```

c) Write the function prototypes (declarations only) for the following (5) situations in C++:

- i. A function add that takes two integers and returns their sum.
- ii. A function display that takes a constant character pointer (string) and does not return any value.
- iii. A function power that takes two integers x and y, where y has a default value of 2, and returns integer.
- iv. A function swap that swaps two integers using call by reference.
- v. A function area that takes a float radius and returns the area of a circle, with a default value of radius as 1.0.

Q6. a) Write a C++ program to count the number of vowels in a file, (5) data.txt.

b) Find the output of the following code:

(2*2.5)

```

i.
#include <iostream>
using namespace std;

template <class T>
class Sample {
    T x;
public:
    Sample(T a) {
        x = a;
    }
    void show() {
        cout << x * 2 << " ";
    }
};

```

```
int main() {
    Sample<int> s1(5);
    Sample<char> s2('A');
    s1.show();
    s2.show();
    return 0;
}
```

ii.

```
#include <iostream>
using namespace std;
int main() {
    try {
        int a = 5, b = 0;
        if(b == 0)
            throw b;
        cout<< a / b;
    }
    catch(char e) {
        cout<< "Character exception\n";
    }
    catch(int e) {
        cout<< "Integer exception\n";
    }
    catch(...) {
        cout<< "General exception\n";
    }

    try {
        throw 'x';
    }
    catch(char e) {
        cout<< "Character exception\n";
    }
    catch(int e) {
        cout<< "Integer exception\n";
    }
    catch(...) {
        cout<< "General exception\n";
    }

    try {
        throw 3.14;
    }
    catch(char e) {
        cout<< "Character exception\n";
    }
    catch(int e) {
```

```

        cout<< "Integer exception\n";
    }
    catch(...) {
        cout<< "General exception\n";
    }
    return 0;
}

```

- c) Write a C++ program to demonstrate constructor overloading using a (5)
class Student with the following specifications:

- i. Data Members: Name, Roll Number, Marks
- ii. Implement the following three types of constructors:
 - a. Default Constructor: Initialize data members with default values
 - b. Parameterized Constructor: Initialize data members with user-defined values.
 - c. Copy Constructor: Initialize an object by copying values from another object
- iii. In the main() function, create objects using all three types of constructors.

- Q7. a) Write a C++ program for a library management system with the (10)
following specifications:

- i. Class: LibraryMember
 - a. Private Data Members: Name of the member, Age
 - b. Public Data Members: Membership ID, Type of membership (Student/Faculty), Number of books issued
- ii. Member Functions (Inside the Class):
 - a. A function to assign initial values
 - b. A function to issue a book (increase book count)
- iii. Member Functions (Outside the Class):
 - a. A function to return a book after verifying that at least one book is issued
 - b. A function to display the member's name and number of books currently issued.
- iv. Create an object of the class and demonstrate the working of the member functions.

- b) Write a program to create a class Car with static data member, number. (5)
Display the number of objects created and destroyed.