

GUJARAT TECHNOLOGICAL UNIVERSITY**MCA – SEMESTER – II • EXAMINATION – SUMMER 2018****Subject Code: 2620002****Date: 21-May-2018****Subject Name: Object-Oriented Programming Concepts & Programming****Time: 10.30 am to 1.00 pm****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1**
- (a)
1. What are the advantages of object oriented programming paradigm? **02**
 2. What is a reference variable? What is its use? **02**
 3. When will you make a function inline? Why? **02**
 4. When do we declare a member of a class static? **01**
- (b)
1. List out the special properties of the constructor function. **03**
 2. What are the different forms of inheritance? Give an example for each. **03**
 3. How many arguments are required in the definition of an overloaded unary operator? **01**
- Q.2**
- (a)
1. Discuss the various forms of get () function supported by the input stream. How are they used? **03**
 2. How is polymorphism achieved at compile time, and at run time? **04**
- (b)
1. Can we have more than one constructor in a class? If yes, explain the need for such a situation. **02**
 2. What is an operator function? Describe the syntax of an operator function. **02**
 3. What will be the output of the following program? **03**
- ```
class Base1 {
public:
 Base1()
 { cout << " Base1's constructor called" << endl; }
};
class Base2 {
public:
 Base2()
 { cout << "Base2's constructor called" << endl; }
};
class Derived: public Base1, public Base2 {
public:
 Derived()
 { cout << "Derived's constructor called" << endl; }
};
int main()
{
 Derived d;
 return 0;
}
```

**OR**

- (b)
1. Why do we need virtual functions? **02**
  2. List out at least four new operators added by C++ which aid OOP. **02**
  3. What will be the output of the following program? **03**

```

class Test
{
public:
 int i;
 void get();
};
void Test::get()
{
 std::cout << "Enter the value of i: ";
 std::cin >> i;
}
Test t; // Global object
int main()
{
 Test t; // local object
 t.get();
 std::cout << "value of i in local t: "<<t.i<<'\n';
 ::t.get();
 std::cout << "value of i in global t: "<<::t.i<<'\n';
 return 0;
}

```

- (A) Compiler Error: Cannot have two objects with same class name  
 (B) Compiler Error in Line “::t.get();”  
 (C) Compiles and runs fine

- Q.3 (a)** 1. What is a friend function? What are the merits and demerits of using friend functions? **03**  
 2. Write a function using reference variables as arguments to swap the values of pair of integers. Demonstrate the use of this function. **04**  
**(b)** What is the use of operator overloading? Explain overloading of prefix ++ and postfix ++ operator with suitable example. **07**

**OR**

- Q.3 (a)** 1. Distinguish between the following two statements: **03**  
     Time T2(T1);  
     Time T2=T1;  
 2. Explain, with an example, how you would create space for an array of objects using pointers. **04**  
**(b)** What is a conversion function? How is it created? Explain with an example. **07**

- Q.4 (a)** 1. What is the difference between manipulators and ios member functions in implementation? Give examples. **03**  
 2. How is an exception handled in C++? When do we use multiple catch handlers? **04**  
**(b)** What is the use of function seekg()? Write the statements using seekg() to achieve the following: **07**  
     1) To move the pointer by 15 position backward from current position.  
     2) To go to the byte number 50 in the file.  
     3) To go backward by 20 bytes from the end.

**OR**

- Q.4 (a)** 1. What is a file mode? Describe the various file mode options available. **03**  
 2. When should a program throw an exception? Explain with an example. **04**  
**(b)** Write a program that creates two files called ODD and EVEN using command **07**

–line argument. Set of numbers stored in an array are written to these files.  
Write odd numbers in the file ODD and even numbers in the file EVEN.  
Display the content of both the files.

- Q.5** (a) 1. What is generic programming? How is it implemented in C++? **03**  
2. List out the types of containers supported by STL and explain any one. **04**  
(b) Demonstrate the use of a template class with two generic data types. **07**
- OR**
- Q.5** (a) 1. What is STL? How is it different from the C++ Standard Library? **03**  
2. What is a namespace conflict? How is it handled in C++? **04**  
(b) Define the general format of a functional template. Demonstrate the use of a function template with an example. **07**

\*\*\*\*\*

[www.FirstRanker.com](http://www.FirstRanker.com)