

Ex/Int/Comp.Sc/8/29/2019(old)

BACHELOR OF SCIENCE EXAMINATION, 2019
(Second Year, First Semester)
COMPUTER SCIENCE (SUBSIDIARY)

Unit-8

(INTRODUCTION TO OBJECT ORIENTED PROGRAMMING USING C++)

Time : 2 hours Full Marks : 50

Attempt Question No. 1 and any *three* from the rest.

1. (a) Find errors with reasons in the following functions:

```
void f1(char* p) {  
    int a = 1;  
    const int c = 2;  
    const int* p1 = &c;  
    const int* p2 = &a;  
    int* p3 = &c;  
    *p3 = 7;  
    const int x;  
}  
void f2(char* p) {  
    double& dr = 1;  
    const double& cdr = 1;  
    int i = 1;  
    int& r1 = i;  
    int& r2;  
    extern int& r3;  
}
```

```
void f3(char* p) {  
    char s[] = "Jadavpur University";  
    const char* pc = s;  
    pc[3] = 'g';  
    pc = p;  
    char* const cp = s;  
    cp[3] = 'a';  
    cp = p;  
    const char* const cpc = s;  
    cpc[3] = 'a';  
    cpc = p;  
}
```

- (b) Explain: argument passing by *value* and *reference*. 5+6 = 11
2. What are the conversions used in the correct expressions with the following program?

```
struct X {  
    int i; X(int); operator+(int);  
};  
struct Y {  
    int i; Y(X);  
    operator+(X);  
    operator int();  
};  
X operator*(X, Y);  
int f(X);  
X x = 1;  
Y y = x;
```

```

int i = 2;

int main() {
    i + 10;      y + 10 * y;  X++;
    x + y + i;  x * x + i;   ++X;
    f(y);      106 + y;     10 + X;
}

```

Modify the program so that it will run correctly and print the values of each expression.

$$4 + 9 = 13$$

3. Design a class to represent a complex number with two *private* data members for the *real* and *imaginary* parts and *friend operator+*; *operator+* member function or operator. Overload the operator + so that the expressions $10.2 + z$, $z + w$ and $z + 10$ can be evaluated, where z and w are complex numbers.

$$5 + 8 = 13$$

4. (a) What are *garbage* and *dangling pointer*?

- (b) Identify and explain the problems due to garbage and dangling pointer with the code:

```

class Name {
    const char* s;
};
class Table {
    Name* p; long sz;
public:
    Table(long s = 15) { p = new Name[sz = s];}
    ~Table() {delete[] p;}
};

```

```
void f(){  
    Table t1, t2 = t1, t3;  
    t3 = t2;  
}
```

Rewrite the code to resolve the problems using *copy constructor* and *copy assignment*. $5 + (4 + 4) = 13$

5. (a) What do you mean by polymorphism in C++?

(b) Consider the following code segment:

```
class Employee {  
    int id; public:  
    void print() {cout << }  
};  
class Manager: protected Employee {int level;};
```

Redesign the classes to introduce the constructors with appropriate initializers which initialize the member variables.

(c) What do you mean by polymorphism in C++. Assuming appropriate constructors along with member initializers, consider the code:

```
class Employee {int id;};  
void main() {  
    Manager mm(4, 3), Employee ee(1) Employee e = m;  
    Employee* pe = &ee;  
    Manager* pm = &ee;  
    pm -> level = 2;  
}
```

Write the output of the program. Identify and explain the use of *polymorphism* in the above implementation. $3 + 8 = 13$

6. (a) Explain *pure virtual* function, and *abstract class* in C++ with examples.

(b) Given the following code in C++

```
class Test {  
    int x;  
public:  
    virtual void show(int) = 0;  
    int getX() { return x; }  
};  
void main() {  
    Test b;  
    return 0;  
}
```

Identify and explain the errors.

$$(4 + 4) + 5 = 13$$

