

W3101 Programming Languages – C++ Midterm Oct 08, 2007

Name:

Student Id:

1. Any program that can be written in C++ can be done in C. Do you agree with the statement? If so, why do we need C++ and what do we gain by using C++ instead of C? If not, describe one program (at a high level) that we can do only in C++ and not in C. Be brief and precise. Explain your ideas clearly. ... (3 marks).
2. Write the constructor code for the executive class given below that takes an integer value called “amount” as input and sets it as the executive’s salary. ... (2 marks).

```
class employee
{
    private:
        int salary;
    public:
        employee(int x) : salary(x) { }
};

class executive : public employee
{
    // Write the constructor code here

}
```

3. What is the output of the following program segment? Please write your answers next (or below) to the functions called in main. ... (4 marks).

```
class baseClass
{
    public:
        baseClass() { };
        virtual void f1()
            { cout << ``In base class f1`` << endl; }
        void f2() { cout << ``In base class f2`` << endl; }
};

class derivedClass
{
    public:
        derivedClass(){ }
        void f1() { cout << ``In derived class f1`` << endl; }
        void f2() { cout << ``In derived class f2`` << endl; }
};

main()
{
    baseClass x;
    derivedClass y;
    baseClass *z = new derivedClass;

    x.f1();

    x.f2();

    y.f1();

    y.f2();

    z->f1();

    z->f2();

}
```

4.

```
class employee
{
    private:
        int salary;
    public:
        employee(..) { /* code */ }
}
```

What are the different ways in which any external (not a member of this class) function or class access the private member of the class employee shown above? You are free to add member functions to the class, if you wish. Write one line code (to show how it is done) for each way you list in your answer. ... (3 marks).

5. Answer the questions given below. ... (3 marks).

a) What is an abstract class? What is its use?

b) How can you make the employee class given below an abstract class?

```

class employee
{
    private:
        int salary;
        int ssn;
    public:
        employee() { ssn = -1; salary = -1; }
        int getSSN { return ssn; }
        int getSalary();
        void
}

```

c) What is the outcome of the code given below?

```

main()
{
    employee x;
    cout << ``SSN: `` << x.getSSN() << endl;
}

```