Object-Oriented Programming, Fall 2008

Final

2:10pm $\sim$ 3:50pm, Wednesday, January 14, 2009

INSTRUCTIONS

1. This is a *closed-book* exam.
2. Try to solve all of the problems.
3. Try to give short answers. (Hint: An answer need not always be longer than the question.)
4. No cheating.
5. Please hand in both the exam sheet and the answer sheet.
6. Please note that unless otherwise stated, all the line numbers for the program listings are for reference only.

1. (20%) Given the following class definition, what special member functions would in some circumstances be implicitly defined by a C++ implementation? What are the names of them?

```
1 struct C {
2     string s;
3 };
```

2. (10%) How the call of a virtual function and the call of a nonvirtual member function are interpreted?
3. (10%) As we discussed in the classroom, a manipulator is an ordinary function called in a non-traditional way. Now, assuming that all required declarations are given, explain—step by step—how the statement `cout << flush;` gets converted to the call `cout.flush();`, as we explained in the classroom.
4. (10%) Draw a picture to show where a global variable, a dynamic variable, and a local variable are allocated, as we discussed in the classroom.
5. (10%) In C++, the operators `operator=`, `operator[]`, `operator()`, and `operator->` must be defined as nonstatic member functions when overloaded. Why?
6. (10%) What would be the output of the following two C++ programs as shown in Listings 1 and 2? Why?

Listing 1: X1.cpp

```
1  #include <iostream>
2
3  using std::cerr;
4  using std::endl;
5
6  class X {
7  public:
8      X() { cerr << "X()" << endl; }
9      ~X() { cerr << "~X()" << endl; }
10 };
11
12 int main()
13 {
14     X a;
15 }
```

Listing 2: X2.cpp

```
1  #include <iostream>
2
3  using std::cerr;
4  using std::endl;
5
6  class X {
7  public:
8      X() { cerr << "X()" << endl; }
9      ~X() { cerr << "~X()" << endl; }
10 };
11
12 int main()
13 {
14     X a();
15 }
```

7. (10%) What is the purpose of template specialization?

8. (10%) What would be the output of the following C++ program? Why?

```cpp
#include <iostream>

using std::cout;
using std::endl;

class B {
public:
    virtual void f(int i)
    {
        cout << "B=" << i << endl;
    }
};

class D : public B {
public:
    void f(short s)
    {
        cout << "D=" << s << endl;
    }
};

int main()
{
    B* p = new D;

    p->f(16);

    delete p;

    return 0;
}
```

9. (10%) What would be the output of the following C++ program? Why?

Listing 3: macro.cpp

```cpp
#include <iostream>

using std::cout;
using std::endl;

#define SQUARE_M(x) x*x

inline int SQUARE_F(int x)
{
    return x*x;
}

int main()
{
    cout << SQUARE_M(3+1*2)*SQUARE_F(3+1*2) << endl;

    return 0;
}
```