CS 300 Data Structures
Problem Set #4 C++ Classes

1. Which of the following is a valid class declaration?
   a) class A { int x; };
   b) class B { }; missing ;
   c) public class A { } java syntax
   d) object A { int x; }; no keyword, object

2. The data members and functions of a class in C++ are by default
   a) protected
   b) private
   c) public
   d) public & protected

3. Wrapping data and its related functionality into a single entity is known as _____
   a) Abstraction
   b) Encapsulation
   c) Polymorphism
   d) Modularity

4. What does polymorphism in OOPs mean?
   a) Concept of allowing overriding of functions
   b) Concept of hiding data
   c) Concept of keeping things in different modules/files
   d) Concept of wrapping things into a single unit

5. Which concept allows you to reuse the written code?
   a) Encapsulation
   b) Abstraction
   c) Inheritance
   d) Polymorphism

6. What will be the output of the following C++ code?

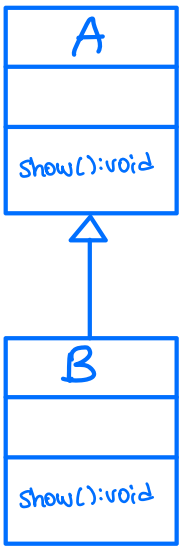
```
#include <iostream>
using namespace std;

class A
{
   int a;
   A()
   {
       a = 5;
   }
};

int main()
{
    A obj();
    cout<< obj.a;
}
```

↑ a is private

a) 0
b) 5
c) Compile-time exception
d) Run-time exception

7. What is the output of the following program?

```
#include<iostream>
using namespace std;

class A
{
    public:
        void show()
        {
            cout<<"A"<<endl;
        }
};

class B: public A
{
    public:
        void show()
        {
            cout<<"B"<<endl;
        }
};

int main(void)
{
    A a;
    a.show();

    B b;
    b.show();

    return 0;
}
```

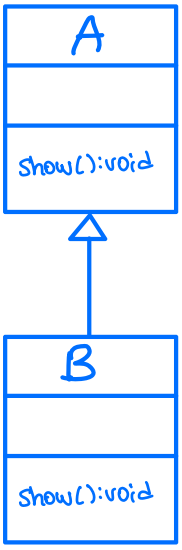


Output: ? A

B

8. Considering the same class A and B definitions in question 7, what will be the output of the following program?

```
int main(void)
{
    A a;
    a.show();

    B b;
    b.show();

    a = b;
    a.show();

    return 0;
}
```



Output:?

A
B
A

9. Considering the same class A and B definitions in question 7, what will be the output of the following program?

```
int main(void)
{
    A a;
    a.show();

    A* aptr = new B;
    aptr->show();

    return 0;
}
```

Output: ?

A
A

STATIC BINDING
COMPILER BINDS FUNCTION AT COMPILE TIME BECAUSE show() IS NOT virtual AND aptr is a A pointer. COMPILER WILL BIND show() FUNCTION IN A CLASS

10. Note the update int virtual function definition update in class A. What is the output of the same program given in question 9?

```
#include<iostream>
using namespace std;

class A
{
    public:
        virtual void show()
        {
            cout<<"A"<<endl;
        }
};

class B: public A
{
    public:
        void show()
        {
            cout<<"B"<<endl;
        }
};

int main(void)
{
    A a;
    a.show();

    A* aptr = new B;
    aptr->show();

    return 0;
}
```

DYNAMIC BINDING BINDS AT RUN-TIME BECAUSE SHOW() IS virtual and overriden in B.

Output: ? A
B