Study Quiz #5

1. Given the classes at the bottom of the page, determine which of the declarations below are valid or invalid. If the declaration is valid, write **YES**, if the declaration is invalid, write **NO.** If it is invalid, explain why on the line below.

    a) _____ `Foo<int, 5> foo1;`

    _____

    b) _____ `Foo foo2(5);`

    _____

    c) _____ `Foo<int, B(5)> foo3;`

    _____

    d) _____ `Foo<A> foo4(B(5));`

    _____

    e) _____ `Foo<B, 5> foo6(5);`

    _____

    f) _____ `Foo<A(), 5> foo7;`

    _____

    g) _____ `Foo<> foo8;`

    _____

    h) _____ `Foo<A, 5> foo10;`

    _____

Classes used in question above

```
template <typename T1 = int, int T2 = 10>
class Foo
{
  public:
    Foo(int x = 0) { }
  private:
    T1 items[T2];
};

class A                              class B
{                                    {
  public:                              public:
    A() { }                              B(int x) : x_(x) { }
};                                       operator int(void)
                                         {
                                             return x_;
                                         }
                                       private:
                                         int x_;
                                     };
```

2.	Given the 4 function prototypes below, which function will be called by the code in function **fn**? Write the letter associated with the function. If the code below is not valid, write **NC**.

```
void foo(char *);    // A
int foo(void);       // B
int foo(int);        // C
double foo(int);     // D

void fn(void)
{
  double d = foo(5);
}


// #1
template <typename T>
T cube(T value)
{
  return value * value * value;
}

// #2
template <>
int cube<int>(int value)
{
  return value * value * value;
}

// #3
int cube(int value)
{
  return value * value * value;
}
```

3.	Given the three functions above, which function will be called by each statement below. Indicate the function by writing the number (**1**, **2**, or **3**) next to the call. If the call is invalid, write **INV**.

a)	_____	`cube<double>(10L);`

b)	_____	`cube<int>(2.5);`

c)	_____	`cube(2.5F);`

d)	_____	`cube(5);`

e)	_____	`cube<char>(5);`

4.	Given a class named Fred, write the declaration (prototype) for its copy constructor.

5.	Given a class named Fred, write the declaration (prototype) for its assignment operator.