1. What is the code below printing?

```cpp
class Top {
    public:
        int x;
        Top(int n) { x = n; }
};
class Left : virtual public Top {
    public:
        int y;
        Left(int m, int n) : Top(m) { y = n; }
};
class Right : virtual public Top {
    public:
        int z;
        Right(int m, int n) : Top(m) { z = n; }
};
class Bottom : public Left, public Right {
    public:
        int w;
        Bottom(int i, int j, int k, int m) : Top(i), Left(i, k), Right(j, k) { w = m; }
};

int main() {
    Bottom b(4, 3, 2, 1);
    cout << b.Left::x << endl;
    cout << b.Right::x << endl;
}
```

2. Write a template class `Vec` for a 2-D vector containing two coordinates of a generic type `A` and with overloaded operators correctly implementing vector **subtraction** and **multiplication by a scalar**.

```cpp
template <class A>
class Vec {
    public:
        A x, y;
        Vec ( A a, A b ) { x=a; y=b; }
        Vec operator- ( const Vec& v2 ) { return Vec(x-v2.x, y-v2.y); }
        Vec operator* ( A d ) { return Vec(x*d,y*d); }
};
```

3. What are the three main differences between pointers and references?

   References a) always have to be initialized, b) cannot be "null", and c) after initialization cannot be changed to reference another object.