Administrivia
PA1.1 assigned todayish (well, more likely tomorrow), due 2/9
Homework review of C++ concepts posted today
Lab posted tomorrow
Object oriented development
1956 – FORTRAN
1972 – C
1982 – C++
1956 – FORTRAN
1967 – Simula 67
1972 – C
1980 – Smalltalk
1982 – C++
Early OO languages were slow
Slow, however, is relative.
Ok, so we’re a lot faster now, why do we still have C/C++?
Code is like diamonds.
This is why software engineering is important!
So what is object oriented development?
Quick review!
Classes and objects
Encapsulation
Aggregation

“Has a”
Inheritance

“Is a”
Inheritance of interface
int readValue(istream & is)
{
    int iValue;
    is >> iValue;
    return iValue;
}

int main(int argc, char * argv[])
{
    int iIntFromConsole = readValue(cin);

    ifstream fs;
    fs.open("file.txt");

    int iIntFromLogFile = readValue(fs);
}
Inheritance of behavior (or implementation)
class Critter {
    public:
        Critter(string sName) {
            m_sName = sName;
            cout << "A " << sName << " appears." << endl;
        }

        virtual ~Critter() {
            cout << "A " << getName() << " departs." << endl;
        }

        virtual string getName() { return m_sName; }

        void eat(Critter * pDinner) {
            cout << "The " << getName() << " eats the " << pDinner->getName() << "." << endl;
            delete pDinner;
        }

    private:
        string m_sName;
    };
class Snake : public Critter
{
 public:
    Snake() : Critter("snake") {}
    virtual string getName() { return "scaly snake"; }
};
Back to Encapsulation
PA1 Part 1: C++-izing the class hierarchy