CS 216

Lecture 2

January 27th, 2014

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.

digital pdp11
digital equipment corporation maynard massachusetts
decpack
dec magtape
Culham
Please do not touch

PDP11- 1970

SAMSUNG
15:06
Thu, 27 Sep
Google
ChatON
Camera
Samsung Apps
Play Store
Phone
Contacts
Messaging
Internet
Apps
27
Newsstand
iTunes
App Store
Game Center
Google Maps
YouTube
Facebook
Settings
Phone
Mail
Safari
Music

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

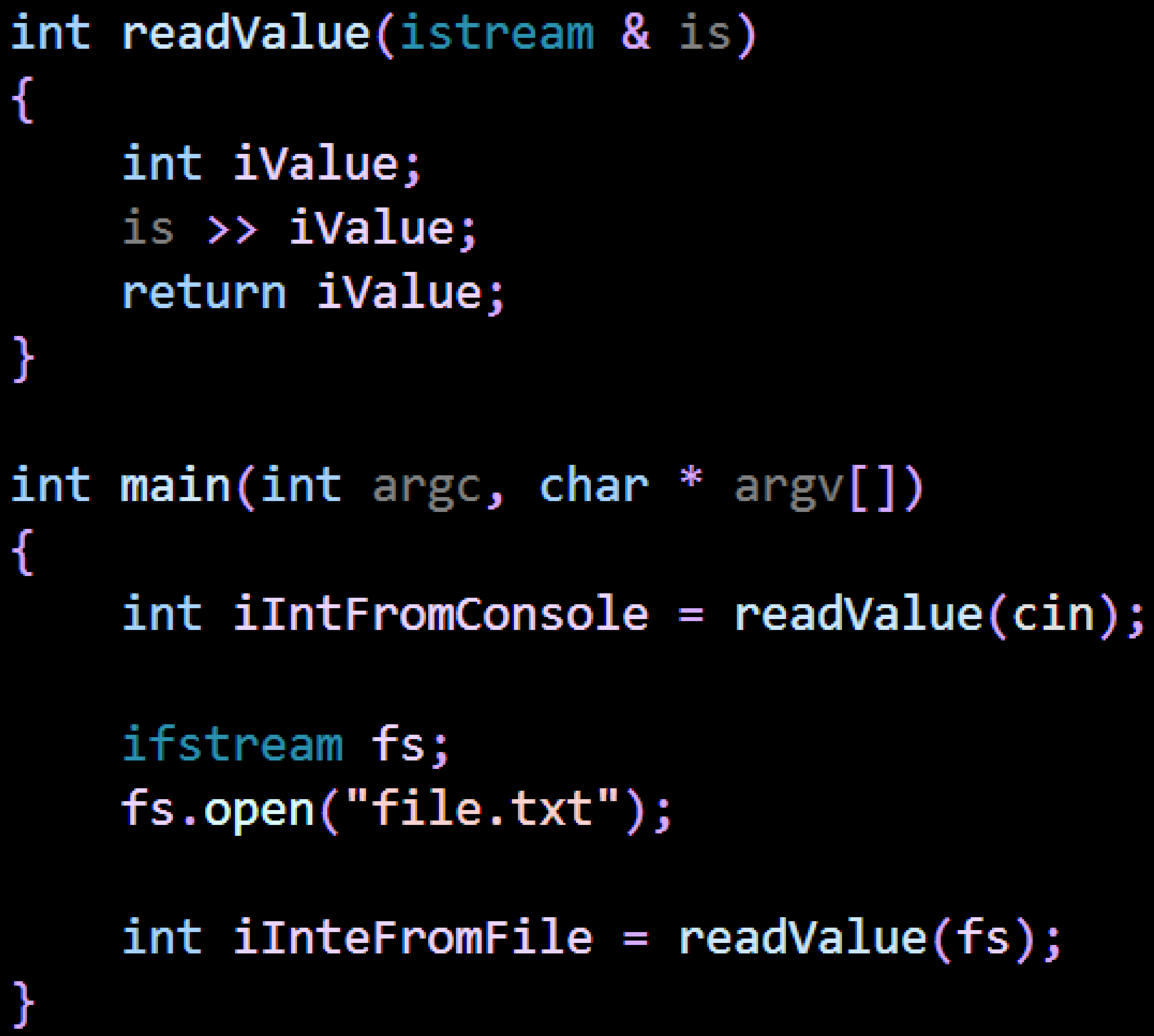
Aggregation

“Has a”

Inheritance

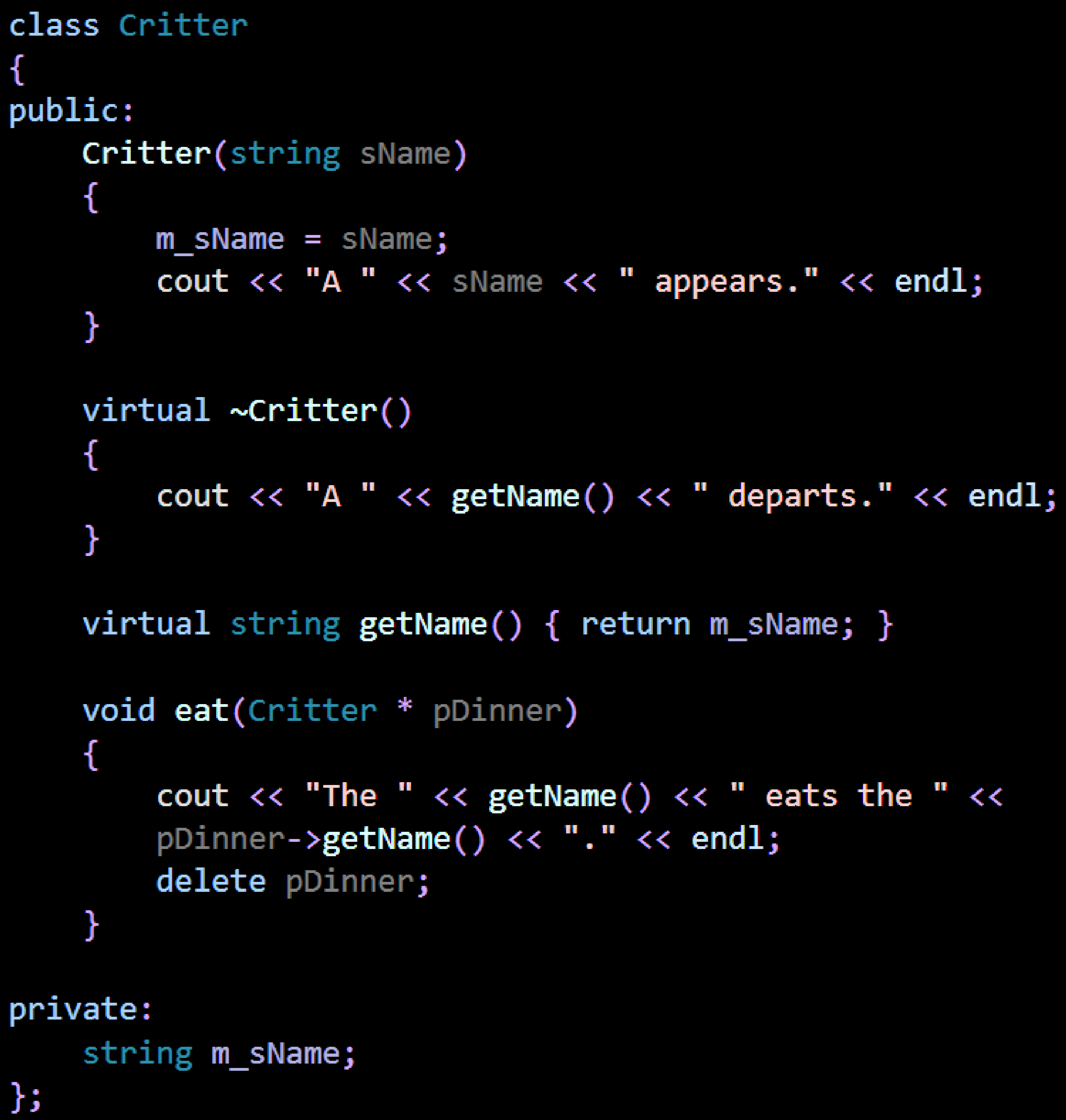
“Is a”

Inheritance of interface

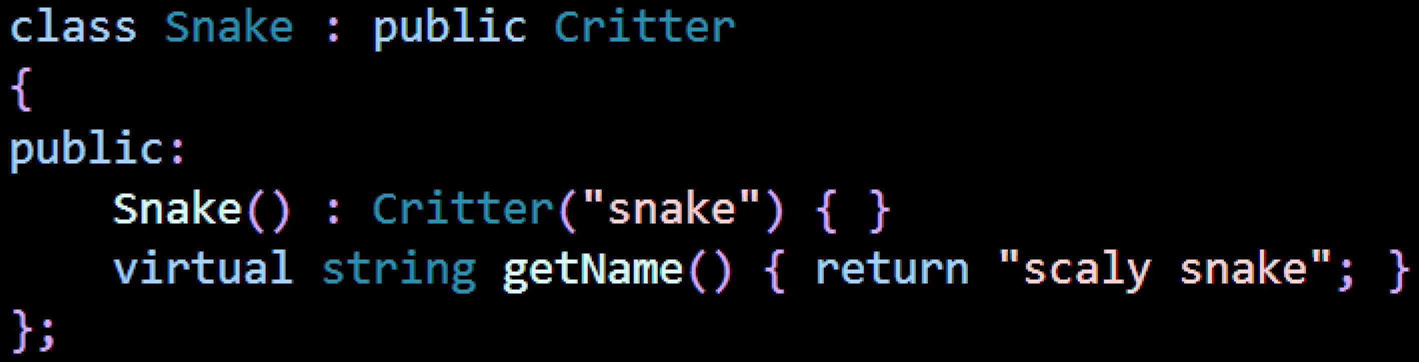


```
int readValue(istream & is)
{
    int iValue;
    is >> iValue;
    return iValue;
}

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

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

    int iInteFromFile = readValue(fs);
}
```

Inheritance of behavior (or implementation)

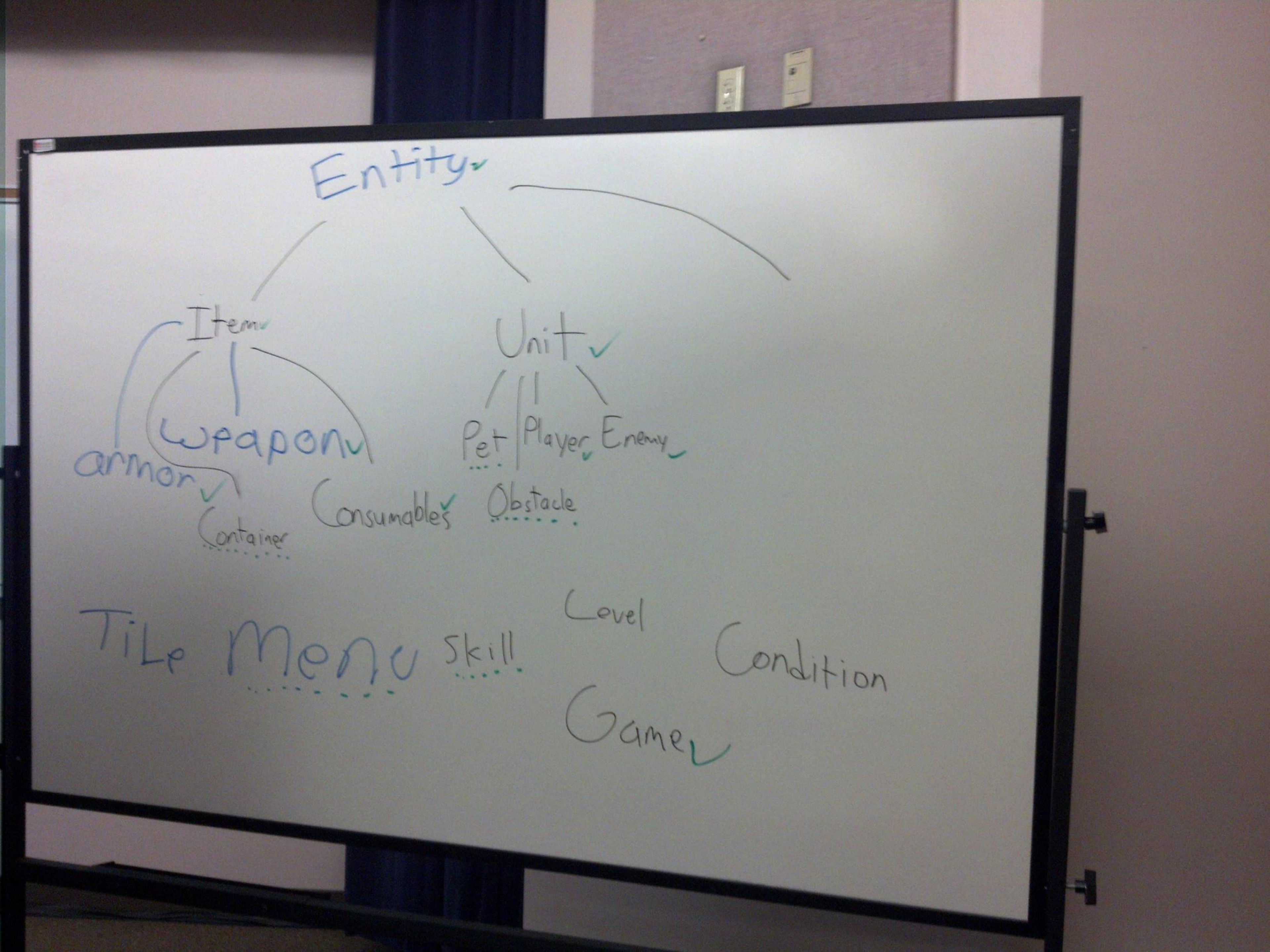


```cpp
class Critter
{
public:
    Critter(string sName)
    {
        m_sName = sName;
        cout << "A " << sName << " appears." << endl;
    }

    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


Entity ✓
Item ✓
Unit ✓
armor ✓
Weapon ✓
Container
Consumables ✓
Pet
Player ✓
Enemy ✓
Obstacle
TiLe
Menu ✓
Skill
Level
Condition
Game ✓

PA1 Part 1: C++-izing the class hierarchy