COS426 Computer Graphics

Precept 0

Jingwan Lu (Cynthia)

February 10, 2012

jingwanl@cs.princeton.edu

Plan for today

- Administrative stuff
- C++ introduction
- Assignment 0

Administrative

I’m Jingwan Lu (Cynthia).
There is also Tianqiang- He uses Mac

My office hour:
418a, Friday 2-3pm
Tianqiang’s office hour:
418a, Monday 3-4pm

Precepts

Currently:

- Fridays, 3pm, CS105

Changes are possible.

Content:

- Supplement to lectures
- Concern assignments and related technicalities

C++

- C++ is an “extension” of the C language. supports “object-oriented programming.”
- Assume you know C and Java

C++

```
#include <iostream>


int main(int argc, char* argv[]))
{
    std::cout << "Hello World!" << std::endl;
    return 0;
}
```

C++

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

int main(int argc, char* argv[]))
{
    cout << "Hello World!" << endl;
    return 0;
}
```

C++

```
#include <iostream>  //<fstream> <string> <vector>
using namespace std;

int main(int argc, char* argv[]))
{
    cout << "Hello World!" << endl;
    return 0;
}
```

C++

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

int main(int argc, char* argv[]))
{
    int n = 0;
    bool m = true;
    string str = “”;

    cin>>n>>m>>str;
    return 0;
}
```

C++ string

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

int main(int argc, char* argv[]))
{
    string      str   = “hello”;
    str               = str + “ world”;           //str = “hello world”
    string      str1 = str.substr(0,  5);        //str1=”hello”
    size_t      pos = str.find(“world”);         //pos=6
    int         l      = str1.length();           //l = 5
    char*       cstr = str.c_str();
    string      str2 = string(argv[0]);
    return 0;
}
```

Reference : http://www.cplusplus.com/reference/string/string/

C++ vector

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

int main(int argc, char* argv[]))
{
    vector<bool>   v1;                //initially empty
    vector<int> v2(4,  0);            //v2 = 0, 0, 0, 0
    v2.push_back(1);                  //v2 = 0, 0, 0, 0, 1
    v2.insert(v2.begin(), 2);         //v2 = 2, 0, 0, 0, 0, 1
    v2.pop_back();                    //v2 = 2, 0, 0, 0, 0
    int     l = v2.size();            //l    = 5
    int     n = v2[0];                //n    = 2
    vector<int> v3 = v2;              // v3 = 2, 0, 0,0, 0
    return 0;
}
```

Reference : http://www.cplusplus.com/reference/stl/vector/

C++

```
#include <iostream>
using namespace std;
const float EPSLON = 0.0001;

int main(int argc, char* argv[]))
{
    const int  NUM = 100;
    int     MyArray1[NUM];                      //Correct
    int     num  = 100;
    int     MyArray2[num];                      //Wrong
    int*    MyArray3 = new int[num];            //Correct,  similar to malloc
    delete []MyArray3;                          //similar to free
    return 0;
}
```

C++ function

```
#include <iostream>

int foo1(int a){
    a = a+1;
    return a;
}
void foo2(int & a){
    a = a+1;
}
void foo3(int *a){
    *a = *a + 1;
}
int foo4(const int & a){
    return a+1;
}
```

```
int main(int argc, char* argv[]))
{
    int a = 0;
    int &b = a;       //b: reference of a
    int *c  = &a;    //c: points to a
    int r;
    r= foo1(a);      //r = 1, a = 0
    foo2(b);         //a = 1
    foo3(c);         //*c=a=2
    r = foo4(a);     //r=3, a=2;
    return 0;
}
```

C++ class

In File Employee.h

```
#ifndef EMPLOYEE_H
#define EMPLOYEE_H
class Employee
{
public:
    Employee();
    Employee(string name,  double salary);
    ~Employee();
    void raise(double x);
    string name() const;
    double salary() const;
private:
    string _name;
    double _salary;
};
#endif
```

In File Employee.cpp

```
#include “Employee.h”

Employee :: Employee()
{ _name = ““;  _salary=0; }

void Employee ::raise(double x)
{
    _salary = _salary + x;
}

string Employee ::name() const
{
    return  _name;
}
```

C++ class

```
#include “Employee.h”

void foo(Employee & p) {p.raise(100);}

int main(int argc, char* argv[]))
{
    Employee  p1;                  //Call the default constructor
    Employee  p2(1, 2);            //Supply construction parameters
    Employee  p3 = p1;             //copies p1 into p3
    p3.raise(100);                 //moves p3 but not p1
    foo(p3);                       //p3._salary = 200, p1._salary = 0
    Employee* p4 = NULL;           //initialize a pointer to NULL
    p4 = new Employee ();          //dynamically create an object
    delete p4;
    return 0;
}
```

C++ Inheritance

```
class Employee
{
public:
    Employee();
    Employee(string name,  double
    salary);
    void raise(double x);
    string name() const;
    double salary() const;
    virtual void print() const
    { cout<<_name<<“ ”<< _salary<<“
    “; }
protected:
    string _name;
    double _salary;
};
```

```
class Manager : public Employee
{
public:
            Manager(string name,
            double salary, string dept);
            void print() const;
private:
            string _department;
};
```

C++ Inheritance

```
class Manager : public Employee
{
public:
    Manager(string name, double salary, string dept);
    void print() const;
private:
    string _department;
};

Manager::Manager(string name, double salary, string dept)
: Employee(name, salary)         // call superclass constructor
{  department = dept;}

void Manager::print() const
{   Employee::print();               //call superclass method
    cout << _department << “ “;}
```

C++ Polymorphism

```
Employee* e1 = new Manager(“Tianqiang Liu”,  22000, “IT”);
Employee* e2 = new Employee(“Cynthia Lu”,  20000);
Vector<Employee*> staff;
staff.push_back(e1);
staff.push_back(e2);

for (i = 0; i < staff.size(); i++){
    staff[i]->print();
    cout<<endl;
}
```

Output:

```
Tianqiang  Liu  22000  IT
Cynthia  Lu  20000
```

Development

Read the COMPILER.txt

- Windows
  - VS 2010 – assignments have .sln files
- Linux
  - Install necessary packages
    - How?
      - http://laptoplogic.com/resources/a-beginners-guide-on-how-to-install-linux-software
- Macs
  - Install XCode IDE

Assignment 0

- R2Image
- R2Pixel
- impro
- Run
- Writeup
- Submission

R2Image

- Access a pixel:
  - Pixel(int x, int y)
- Set a pixel:
  - SetPixel(int x, int y, const R2Pixel& pixel)

Tip: Check the .h file before you want to implement any functionality – much of it is already there!

R2Pixel

R2Pixel pixel:

- Four Channels (R,G,B, Alpha)
- Component access
  - x = pixel.Red() ⇔ x = pixel[0] ⇔ x = pixel.Component(0)
- Set components
  - pixel.SetGreen(0.5) ⇔ pixel[1] = 0.5
- pixel.Clamp()

Tip: Check the .h file before you want to implement any functionality – much of it is already there!

HTML for the write-ups

- Layout
- Tags
- More elaborate tutorial
  - http://www.amittai.com/prose/tutorial.html

Assignment 0

- Show
  - Impro
  - Running
    - Cygwin/Linux/Mac
    - VS command prompt
  - Writeup
- Submission
  - Preserve directory tree
  - ZIP files
  - Naming zip file: johndoe_cos426_assignment0.zip
  - Submit via Dropbox