CSCI-1411 Fundamentals Of Computing Lab

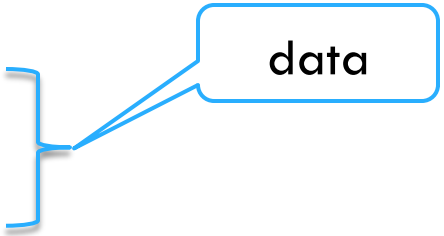
Anh Nguyen

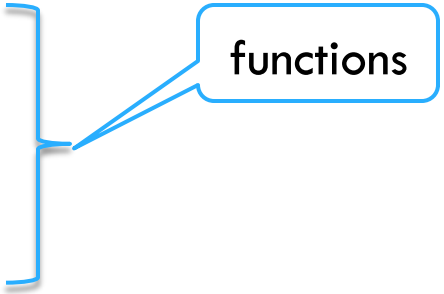
Fall 2015

Lab 13: Introduction to Classes

- Class: ADT – a prototype (template) for a set of objects
- → both functions and data are encapsulated (contained)
- → functions and data can be accessed privately or publicly

```
class Rectangle
{
private:
      float length;
      float width;
public:
      void setLength(float side_l);
      float getWidth();
      double findArea();
      …
};
```





Lab 13: Introduction to Classes

- Implementations of a class in C++
  - Class declaration → a header (.h) file
  - Member functions of a class → an implementation (.cpp) file
- Implementation of member functions
  - Using the :: symbol (or scope operator)
  - Method 1: In the implementation file

```
function_datatype class_name::function_name(function's arguments)
{
…
}
```

  - Method 2: Inline member functions

```
double Rectangle::findArea() { return length * width; }
```

Lab 13: Introduction to Classes

- Constructor: initialization of an object
  - Be implicitly invoked whenever a class instance is created
  - Have the same name as the class itself
  - Do not have a data type (or the word void) in front of it

```
class Rectangle
{
private:
        float length;
        float width;
public:
        Rectangle();
        Rectangle(float side_l, float side_w);
        …
};
```

```
Rectangle::Rectangle()
{
        length = width = 1;
}

Rectangle::Rectangle(float side_l, float side_w)
{
        length = side_l;
        width = side_w;
}
```

Lab 13: Introduction to Classes

- Destructor: to destroy an object
  - Be preceded by a tilde (~)
  - Free up memory when the object is no longer needed
  - Be automatically called when an object goes out of scope

```
class Rectangle
{
private:
        float length;
        float width;
public:
        Rectangle();
        Rectangle(float side_l, float side_w);
        ~Rectangle();
        …
};
```

constructors

destructors

Lab 13: Introduction to Classes

- Instance of a class: an object whose data and functions are accessed using the dot operator.

```
void main
{
    Rectangle rBox1;
    Rectangle rBox2(2, 4);
    rBox1.setLength(3);

    double rArea = rBox2.findArea();

    Rectangle aBox[100];
}
```

instance

dot operator

Lab 13: Introduction to Classes

- 13.1Squares as a Class
  - (square.cpp)
  - Complete Exercise 2
- 13.2 Circles as a Class
  - (cirlces.cpp)
  - Skip Exercise 3
  - Implement Exercise 4
  - Answer the exercise 1 & 4
- 13.3 Arrays as Data Members of Classes
  - (floatarray.cpp)
  - Temperatures.txt
  - Answer the exercise 1

Lab 13: Introduction to Classes

- 13.4 Arrays of Objects
  - (inventory.cpp)

- 13.5 Code Assignment
  - No Options (complete Exercise 1 and 2)
  - Name the source file (main.cpp)

Lab 13: Introduction to Classes

- Submission File Checklist
  - Submit all files on Canvas (One at a time or all of them in a single zip file). Be sure to include all source files and documents.

  - 13.1 square.cpp
  - 13.2 circles.cpp
  - 13.3 floatarray.cpp
  - 13.4 inventory.cpp
  - 13.5 main.cpp