CSI33 Data Structures

Sharon Persinger

Fall 2019

Day 24 November 27

Topics

Errors in Dynamic Memory Allocation with Linked Lists

Template Functions

Template Classes

Template functions

In C++ you can write functions that work for any type, as long as the operations used in the function definition are defined for the type.

Example: a function that returns the maximum of two comparable items a, b

For ints:

```
int maximum(int a, int b){
If (a > b) {return a;}
else {return b;}
}
```

The function is the same for doubles except for the typenames.

Template version:

```
template <typename Item> //Item stands for
                         //name of a type

Item maximum(Item a, Item b){

if (a > b)
        {   return a;  }

else

        {   return b;  }

}
```

Run the program maximum.cpp

Details of template functions

Syntax: template <typename Item> preceeds the function prototype.

template and typename are key words.

Item is an alias for an typename – a built-in type, a class, or an array.

The compiler generates the machine instructions for a template function when the function is called. This is called instantiation of the function. The specific machine instructions used depend on the data type. At the machine level, doubles are compared in a different way from ints, for instance.

For this function, when the function is called, each variable of type Item must be of the same type.

Template classes

Template classes work in the same fashion.

template <class T> precedes the class definition. T is the stand-in for a typename

Look at Mypair.cpp

C++ Standard Template Library

C++ Standard Template Library (STL) has template versions of many different classes.

pair: 2-tuple

vector: dynamically allocated array

list: doubly-linked list

slist: singly linked list

queue: FIFo queue

Priority queue: like a queue but items of higher priority are served before items of lower priority

stack: LIFO stack

set: mathematical set

Using classes from the STL

Example with vector class

vectorexamples.cpp

Defining a template class

Example: Write a template definition of a Stack class.

You can put the class declaration and all of the method definitions into the header file.

It is also common to place the class declaration into a header file with extension .h and put the implementation into a template file with extension .template. Then the .template file is included in the header file, at the end of the file. Our example uses this approach.

Stack.h

Stack.template