ET-580 – Templates & Exceptions - Homework

Reading

Submit Notes: **Chapter 17 Linked Data Structures**

Implementation

Implement a class SomeObj:

1) data member:
   a) id - an integer named
2) constructors:
   a) default
   b) single parameter constructor to set the value of id
3) functions:
   a) getId - return the value of id
   a) output - output the value of id to console

Implement a Template Class MyArray (modify/augment example 12.6):

1) data members:
   a) data - dynamic array of type T
   b) capacity - capacity of the dynamic array
   c) size - number of elements in the array
2) constructors:
   a) default - create an empty array of capacity 10
   b) single parameter - create an array of a specified capacity
3) functions:
   a) getSize - return the size
   b) getCapacity - return the capacity
   b) grow - private grow function (capacity*2)+1
   c) getIndex - given a value, return its index or -1 if not found
      use a linear or binary search
   d) push_back - append element to the end of the array
   e) erase - remove the element at the specified index
   f) [] - overloaded [] operator (regular and const versions)
4) the big three

A Non-Member Print function:

1) Prints the contents of a template *MyArray* object given via as a parameter.

Main:

1) Instantiate a template *MyArray* object of *SomeObj* pointers.
2) Instantiate 10 *SomeObj* objects and append them to the MyArray object.
3) Print the SomeObj contents of the MyArray object.

Some additional information:

getIndex function (using linear search):
1) Accepts a value of type T as a parameter.
2) Iterate through the array checking if the value is in the array.
   If it is return the index.
   If it is not, return -1.

Example

Given ***5 10 15 20 25 30*** and value ***15***, return the index ***2***.

erase function:
1) Accepts an index as a parameter.
2) Use a loop to shift all elements from the index to the end of the array to the left by 1.
3) Decrement size by 1.

Example

Given ***5 10 15 20 25 30*** and index ***3*** we would shift ***25*** and ***30*** to the left thereby overwriting the number ***20*** resulting in the array ***5 10 15 25 30***.