WAYNE STATE UNIVERSITY
COLLEGE OF ENGINEERING

Computer Science Department

CSC 2201: Computer Science II – Lab
Lab6

Description:

You will implement your own Stack ADT, the StackArray & StackLinked class.

Goals:

Learn how to use C++ classes to implement Stack abstract data type (ADT).

Where to Start:

1. Download Lab6.zip from Blackboard
2. Unzip the file
3. Open the solution in Microsoft Visual Studio
4. Start Implementing the operations of the StackArray class:

   4.1. Implement default constructor: `StackArray<DataType>:: StackArray (int maxNumber )`

   4.2. Implement copy constructor: `StackArray <DataType>:: StackArray (const StackArray &other)`

   4.3. Implement operator =: `StackArray <DataType>& StackArray <DataType>::operator = ( const StackArray &other)`

   4.4. Implement destructor: `StackArray <DataType>:: ~ StackArray ()`

   4.5. Implement push (): `void StackArray <DataType>:: push ( const DataType &newDataItem ) throw ( logic_error )`

   4.6. Implement pop (): `DataType StackArray <DataType>:: pop () throw (logic_error)`

   4.7. Implement clear (): `void StackArray <DataType>:: clear ()`

   4.8. Implement isEmpty(): `bool StackArray <DataType>:: isEmpty () const`

   4.9. Implement isFull(): `bool StackArray <DataType>:: isFull () const`

   4.10. Implement showStructure (): `void StackArray <DataType>:: showStructure ()const`

5. Start Implementing the operations of the StackLinked class:

   4.1. Implement default constructor: `StackLinked<DataType>:: StackLinked (int maxNumber )`

   4.2. Implement copy constructor: `StackLinked <DataType>:: StackLinked (const StackLinked &other)`

   4.3. Implement operator =: `StackLinked <DataType>& StackLinked <DataType>::operator = ( const StackLinked &other)`

   4.4. Implement destructor: `StackLinked <DataType>:: ~ StackLinked ()`

4.5. Implement push (): `void StackLinked <DataType>:: push ( const DataType &newDataItem ) throw ( logic_error )`

4.6. Implement pop (): `DataType StackLinked <DataType>:: pop () throw (logic_error)`

4.7. Implement clear (): `void StackLinked <DataType>:: clear ()`

4.8. Implement isEmpty(): `bool StackLinked <DataType>:: isEmpty () const`

4.9. Implement isFull(): `bool StackLinked <DataType>:: isFull () const`

4.10. Implement showStructure (): `void StackLinked <DataType>:: showStructure ()const`

6. Compile your implementation of the TWO Stack ADT.
7. Test your implementation using the program in the file test6.cpp.
8. The output would be like the following:

```
Commands:
  H  : Help (displays this message)
  +x : Push x
  -  : Pop
  C  : Clear
  E  : Empty stack?
  F  : Full stack?
  Q  : Quit the test program

Empty stack

Command: +a
Push a
Top     [a]     Bottom

Command: +b
Push b
Top     [b]     a       Bottom

Command: +c
Push c
Top     [c]     b       a       Bottom

Command: -
Popped c
Top     [b]     a       Bottom

Command: e
Stack is NOT empty
Top     [b]     a       Bottom

Command: f
Stack is NOT full
Top     [b]     a       Bottom

Command: c
Clear the stack
Empty stack
```

StackArray

```
Testing ARRAY implementation

Commands:
  H  : Help (displays this message)
  +x : Push x
  -  : Pop
  C  : Clear
  E  : Empty stack?
  F  : Full stack?
  Q  : Quit the test program

Empty stack.


Command: +a
Push a
top = 0
0        1        2        3        4        5        6        7
[a]


Command: +b
Push b
top = 1
0        1        2        3        4        5        6        7
a        [b]


Command: +c
Push c
top = 2
0        1        2        3        4        5        6        7
a        b        [c]


Command: -
Popped c
top = 1
0        1        2        3        4        5        6        7
a        [b]


Command: e
Stack is NOT empty
top = 1
0        1        2        3        4        5        6        7
a        [b]


Command: f
Stack is NOT full
top = 1
0        1        2        3        4        5        6        7
a        [b]


Command: c
Clear the stack
Empty stack.
```

StackLinked

Create a Zip file of your solution:

1. Right click on your solution in Solution Explorer
2. Click on "Open Folder in File Explorer"
3. Go one level up in file explorer
4. Right click on your solution folder
5. Add it to archive by creating a zip file

Upload the zipped file on Blackboard:

1. Go to Blackboard
2. Click on this course (CSC 2201: Computer Science II Lab)
3. Go to the folder "Labs"
4. Click on the "Lab4_Work" assignment
5. Upload your zipped file