Amy Jiang
5/15/2017

Homework #7 (Due 05/18/17 at 11:55 pm)

#1. Write the C++ (not just pseudocode) for a client program that creates a stack of the strings "Jamie", "Jane" and "Jill" in that order with "Jamie" at the top of the stack, "Jane" next and "Jill" at the bottom. Peek at each item and then pop it off the stack to show Jamie is at the top, Jane next and Jill at the bottom. (10 points)

```
#include <iostream>
#include <string>
#include "ArrayStack.h"
using namespace std;

int main() {
	ArrayStack<string> nameStack;
	nameStack.push("Jill");
	nameStack.push("Jane");
	nameStack.push("Jamie");
	for (int i = 0; i < 3; i++) {
		cout << nameStack.peek() << endl;
		nameStack.pop();
	}

	system("PAUSE");
	return 0;
}
```

Amy Jiang
5/15/2017

#3a. Suppose you have a stack aStack and an empty extra stack called extraStack. Give pseudocode for accomplishing the following task:. Display the contents of aStack in reverse order: so display the contents from bottom to top (display the bottom item first, then the second to the bottom item and so on until you get to the top). Note: you may want to use the extraStack in your algorithm. (10 points)

```
#include <iostream>
#include <string>
#include "ArrayStack.h"
using namespace std;

int main() {
	ArrayStack<string> aStack;
	ArrayStack<string> extraStack;
	aStack.push("A");
	aStack.push("B");
	aStack.push("C");
	while (!aStack.isEmpty()) {
		extraStack.push(aStack.peek());
		aStack.pop();
	}
	while (!extraStack.isEmpty()) {
		cout << extraStack.peek() << endl;
		aStack.push(extraStack.peek());
		extraStack.pop();
	}
	system("PAUSE");
	return 0;
}
```

Amy Jiang
5/15/2017

#3b. Suppose you have a stack aStack and an empty extra stack called extraStack. Give pseudocode for accomplishing the following task:. Count the number of items in aStack but do NOT remove any items from aStack, it must remain in tact, unchanged. Again you may want to use extraStack in your algorithm. (10 points)

```
#include <iostream>
#include <string>
#include "ArrayStack.h"
using namespace std;

int main() {
	ArrayStack<int> aStack;
	ArrayStack<int> extraStack;
	int count = 0;
	aStack.push(1);
	aStack.push(2);
	aStack.push(3);
	aStack.push(4);
	while (!aStack.isEmpty()) {
		extraStack.push(aStack.peek());
		aStack.pop();
		count++;
	}
	while (!extraStack.isEmpty()) {
		aStack.push(extraStack.peek());
		extraStack.pop();
	}
	cout << "The number of items in aStack is: " << count << endl;
	system("PAUSE");
	return 0;
}
```

Amy Jiang
5/15/2017

#3c. Suppose you have a stack aStack and an empty extra stack called extraStack. Give pseudocode for accomplishing the following task:. Remove every occurrence of a specified item from aStack, leaving the order of the remaining items in aStack unchanged. Again you may want to use extraStack in your algorithm. (10 points)

```
#include <iostream>
#include <string>
#include "ArrayStack.h"
using namespace std;

int main() {
	ArrayStack<int> aStack;
	ArrayStack<int> extraStack;
	int spItem;
	aStack.push(1);
	aStack.push(2);
	aStack.push(3);
	aStack.push(3);
	aStack.push(3);
	cout << "enter the number you wish to remove: " << endl;
	cin >> spItem;
	while (!aStack.isEmpty()) {
		if (aStack.peek() != spItem)
			extraStack.push(aStack.peek());
		aStack.pop();
	}
	while (!extraStack.isEmpty()) {
		//cout << extraStack.peek() << " ";
		aStack.push(extraStack.peek());
		extraStack.pop();
	}
	system("PAUSE");
	return 0;
}
```