

## SWE315 : C++

**Homework 16 (Real sequence is #9) - ( Due date 08-14 : Doing most of it in class!!)****General notes:**

1. Please send solution to: [zbaharav@cogswell.edu](mailto:zbaharav@cogswell.edu)
2. You know the drill by now: Simply hit reply, and no zipped directories etc..
3. Please attach this document with your solution (including screen shots!) at the bottom.

====

**Assignment:**

(This assignment is based on: "C++ in plain English", by Brain Overland).

Start with the following Stack implementation (we wrote all this in class):

```

/* File main.cpp */
#include <iostream>

class Stack{
private:
    int *pStack;
    int size;
    int index; // points to the next empty spot

public:
    // constructor
    Stack(int sz){ pStack = new int[size = sz]; index=0; }
    // destructor
    ~Stack() { delete [] pStack;}

    // utility
    int pop(void) { return pStack[--index]; }
    void push(int item) { pStack[index++] = item; }
};

int main(void)
{
    std::cout << "Starting program: Stack\n" ;

    Stack S1(20);

    // Push in
    std::cout << "Push in two values: 5 and 7. \n";
    S1.push(5);
    S1.push(7);

    // pop out
    std::cout << "Pop out two values: " ;
    std::cout << S1.pop() << " and " ;
}

```

```

std::cout << S1.pop() << " .\n" ;

system("pause");
return 0;
}
/* End of File main.cpp */

```

Your task will be to turn this into a template container class, improve a few things, and test it.

1. Convert the class to be template based, with the type of the elements in the stack as the template variable.
2. Replace your main() function with the one given below.
3. Try and run your program with the /\* 2nd part\*/ of main below commented.
4. Make the following improvements, by moving functions OUTSIDE of the class definition (just declare them there, and implement outside the class template)
  - a. Pop function: if there are no items in the stack, return a 'dummy' object (of the type of the stack). That means just uninitialized empty object.
  - b. Push function: make it return an integer. If the stack is full, returns 0. Otherwise, return 1.
5. Test with /\*2<sup>nd</sup> part\*/ uncommented.

See expected result below!! (screen shot)

```
/* Testing main function (replace with the main function you had above) */
```

```

/* you will also need to include
#include <string>
*/

int main(void)
{
    std::cout << "Starting program: Stack\n" ;

    // Testing template int.
    std::cout << "\nDeclaring Stack<int>\n" ;
    Stack<int> S1(20);

    // Push in
    std::cout << "Push in two values: 5 and 7. \n";
    S1.push(5);
    S1.push(7);

    // pop out
    std::cout << "Pop out two values: " ;
    std::cout << S1.pop() << " and " ;
}

```

```

std::cout << S1.pop() << " .\n" ;

// Testing template string.
std::cout << "\nDeclaring Stack<string>\n" ;
Stack<std::string> Str1(10);

Str1.push("A string");
Str1.push("B string");

std::cout << "Pop out two values: " ;
std::cout << "\"" << Str1.pop() << "\" and " ;
std::cout << "\"" << Str1.pop() << "\" .\n" ;

/* 2nd part: Please keep commented until instructed! */
/*
// Testing template string with checks...
std::cout << "\nDeclaring Stack<string> of size 2\n" ;
Stack<std::string> Str2(2);

std::cout << "\nTrying to pop an empty stack returns: " << Str2.pop() <<
std::endl;

std::cout << "\nTrying to push four items.\n" ;

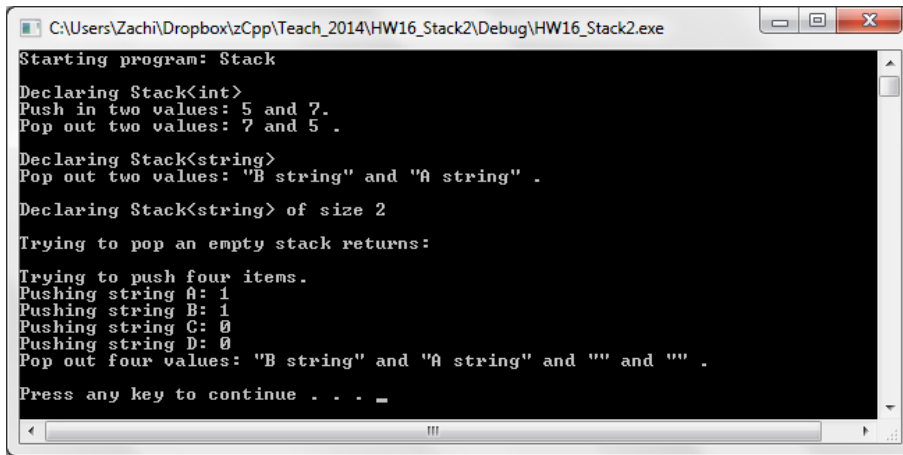
std::cout << "Pushing string A: " << Str2.push("A string") << std::endl;
std::cout << "Pushing string B: " << Str2.push("B string") << std::endl;
std::cout << "Pushing string C: " << Str2.push("C string") << std::endl;
std::cout << "Pushing string D: " << Str2.push("D string") << std::endl;

std::cout << "Pop out four values: " ;
std::cout << "\"" << Str2.pop() << "\" and " ;
std::cout << "\"" << Str2.pop() << "\" and " ;
std::cout << "\"" << Str2.pop() << "\" and " ;
std::cout << "\"" << Str2.pop() << "\" .\n\n" ;

*/

system("pause");
return 0;
}
/* End of function main() to use */

```



```
C:\Users\Zachi\Dropbox\zCpp\Teach_2014\HW16_Stack2\Debug\HW16_Stack2.exe
Starting program: Stack
Declaring Stack<int>
Push in two values: 5 and 7.
Pop out two values: 7 and 5 .

Declaring Stack<string>
Pop out two values: "B string" and "A string" .

Declaring Stack<string> of size 2
Trying to pop an empty stack returns:

Trying to push four items.
Pushing string A: 1
Pushing string B: 1
Pushing string C: 0
Pushing string D: 0
Pop out four values: "B string" and "A string" and "" and "" .

Press any key to continue . . . _
```

=== End of Homework 14 ===