# Assignment 3 for CPTN278

**Name:** Stack Implementation with Dynamically Linked Pointers

**References:**

* The text book
* In class notes
* Any other resource you can find other than people

### Assignment Description:

You are to create an application in C++ that uses a Stack Data Structure. You will also be required to provide a detailed description of the program along with conclusions.

The application you create for Assignment 3 is IDENTICAL to the application you created for Assignment 2. That is, the application source file for Assignment 3 MUST be identical to the application source file from Assignment 2 except for the include file references and comments.

Details on program format and construction are provided in class.

### Assignment Deliverables:

1. A C++ class header file containing your class data members and member function definitions. The base name of this file must be **CPTN278\_A3\_Stack\_*lastname*.h** where *lastname* is your actual last name.
2. A C++ class body source file containing your class source code. The base name of this file must be **CPTN278\_A3\_Stack\_*lastname*.cpp** where *lastname* is your actual last name.
3. A C++ application program source file containing your application source code. The base name of this file must be **CPTN278\_A3\_Application\_*lastname*.cpp** where *lastname* is your actual last name.
4. A Microsoft Word 2010 document with the following contents and sections. Each major bullet needs to be a new section. The name of this file must be **CPTN278\_A3\_Description\_*lastname*.docx** where *lastname* is your actual last name.

* Title page
  + Document name
  + Author
  + Creation Date
  + Course Number
* Table of Contents
* Introduction
* A one paragraph description of what the application does
* A description of the program variables and objects
* A description of the class and application source code including
  + The list and correct sequence of steps used to accomplish the task
  + A description of what occurs in a section of code
  + A description of how the variables and objects “change” as a result of executing each section of code
* Captured Program Output
* Conclusions about the application
* Conclusion
* References documented using APA style
* Appendix (if needed)

### Strenuously Recommended Approach:

Even though the word “recommended” is in this section header, consider it as mandatory. The approach below includes requirements and “how to” suggestions that should be used to complete the assignment. Hopefully by following the steps in order and using the suggestions, completing the assignment won’t be a difficult task.

1. Understand the problem.

See Assignment 2.

1. Identify and gather resources.

See Assignment 2.

1. Start your documentation.

I suggest you reuse the document from Assignment 2. Just remove the parts not required for this assignment and keep those you do need. Change the headings and titles as needed. You can then fill it in as you go.

1. Add the document introduction.

See Assignment 2.

1. Write an overview of what the application does.

See Assignment 2.

1. Add the description of the applications variables and objects.

See Assignment 2.

1. Add the source code design for the stack class body.

See Assignment 2.

1. Add the source code design for the application.

See Assignment 2.

1. Start the application source program and stack class.

Actually, just copy the application source file from Assignment 2. Change the include file references and comments. Comment out all the unnecessary code.

For the stack class, I suggest you copy the stack class files from Assignment 2. Change the include file references and comments. Then comment out all the unnecessary code.

Don’t forget to use the correct file names for these files.

This should leave you with a set of “empty” files. Verify these files can be compiled, linked and run. You full application infrastructure is ready to go.

1. In a repetitive process add one small feature to the application and test it.

See Assignment 2.

For this assignment, this is usually accomplished by un-commenting code in the application source file and replacing the stack class appropriate member function code . Remember, do this in small steps.

1. Run the final application and capture the screen output.

See Assignment 2.

1. Document your application conclusions.

See Assignment 2.

1. Add the document conclusion.

See Assignment 2.

1. Build the references page from the in-line citations.

See Assignment 2.

1. Proofread your Word document.

See Assignment 2.

1. Rebuild the TOC.

See Assignment 2.

1. Post the deliverables to Blackboard via the “View/Complete Assignment” link.

You will be submitting four files. They are:

* CPTN278\_A3\_Stack\_lastname.h – The stack class header file
* CPTN278\_A3\_Stack\_lastname.cpp – The stack class body file
* CPTN278\_A3\_Application\_lastname.cpp – The application source file
* CPTN278\_A3\_Description\_lastname.docx – Your design document

Use the above link for assignment submissions. For those not familiar with this Blackboard method of assignment submission, it will be demonstrated in class. Do not submit assignments via the Blackboard Digital Drop Box or forums. Do not use e-mail attachments. Do not hand in hard copies of documents. Do not compress them, zip them or alter their format in any way. This is the only acceptable submission method.