

CSCE 221 Cover Page  
Programming Assignment #1  
Due February 5 by midnight to CSNet

First Name: Shaeeta      Last Name: Sharar      UIN: 822006676

User Name: ssharar      Email address: ssharar@tamu.edu

Please list all sources in the table below including web pages which you used to solve or implement the current homework. If you fail to cite sources you can get a lower number of points or even zero. According to the University Regulations, Section 42, scholastic dishonesty are including: acquiring answers from any unauthorized source, working with another person when not specifically permitted, observing the work of other students during any exam, providing answers when not specifically authorized to do so, informing any person of the contents of an exam prior to the exam, and failing to credit sources used. Disciplinary actions range from grade penalties to expulsion read more: Aggie Honor System Office

Type of sources		
People	PT: Cody Taylor	PT: Wes Tang
Web pages (provide URL)		
Printed material	Textbook: Data Structure and Algorithms in C++	Textbook: Programming Principles and Practice using C++
Other Sources	Slides: Analyzing Algorithms	Slides: Weekly Reading: Analyzing and Writing Algorithms

I certify that I have listed all the sources that I used to develop the solutions/codes to the submitted work.

“On my honor as an Aggie, I have neither given nor received any unauthorized help on this academic work.”

Your Name

Date

## Programming Assignment 1 (130 points)

In the first phase of the assignment, implement in C++ a class `My_vec`, that can hold data of character type. The description of the functions for data manipulation is provided in the first set of the lecture notes slides, see the slide 8. In the second phase, write a generic version of the class `My_vec` that can handle any type of data.

### Instructions

1. Download the supplementary file with a sample code from the class webpage.
2. Your files should be arranged as below
  - (a) Declaration of `My_vec` class in `My_vec.h`
  - (b) Definition (implementation) of `My_vec` class in `My_vec.cpp`
  - (c) Testing code in `Main.cpp`
  - (d) Use the `Makefile`
3. Compile your program using the Linux machine command line:  

```
g++ -std=c++11 *.cpp
```

or  

```
make all
```
4. Run your program by executing  

```
./Main
```
5. Be sure to increase or decrease "size" when you insert or remove from the vector.
6. Be sure to check vector size against its capacity. If size is greater than capacity allocate more memory by doubling the current capacity.

### Points Distribution for Assignment

1. `My_vector` Class Member Functions:
  - (a) (4 pt) `elem_at_rank`
  - (b) (8 pt) `insert_at_rank`
    - i. (4 pt) `replace_at_rank`
    - ii. (8 pt) `remove_at_rank`
    - iii. (10 pt) constructor and copy constructor
    - iv. (10 pt) destructor and the assignment operator
    - v. (6 pt) overloading the `[]` operator

- (c) non-member functions:
    - i. (6 pt) overloading `<<` operator (output operator)
    - ii. (10 pt) `find_max_index` - an index of the largest object in a vector
    - iii. (10 pt) sort `My_vector` using the function `find_max_index`
  - (d) (14 pt) Testing program (main function)
  - (e) (5 pt) Programming style: naming, indentation, whitespace, comments, declaration, variables and constants, expressions and operators, line length, error handling and reporting. Please refer to the document <http://www.stroustrup.com/Programming/PPP-style.pdf>
2. Write a generic version of `My_vector`
    - (a) (15 pt) Test all the operations listed above using at least three different type of objects.
  3. Typed report (preferably using "Lyx/L<sup>A</sup>T<sub>E</sub>X")
    - (a) (2 pt) Program Description; Purpose of the Assignment
    - (b) (4 pt) Data Structure Description and ADT Run-time of the Class `My_vector`:  
Compare ADT run-time for your implementation with the theoretical run-time.  
Provide analysis of best and worst scenarios for `My_vector` ADT.
    - (c) (2 pt) Instructions to Compile and Run your Program; Input and Output Specifications
    - (d) (2 pt) Logical Exceptions (And bug description)
    - (e) (5 pt) C++ object oriented or generic programming features, C++11 features.
    - (f) (5 pt) Testing results

### Submission

1. Tar the four files above and any extra files. Please create your tar file following the instructions at:  
<http://courses.cs.tamu.edu/teresa/csce221/html/tar-file.html>.
  - (a) "turnin" your tar file to the CSNET no latter than February 5 by midnight.
2. Submit a hard copy of cover page, report (see above), `My_vec.h`, `My_vec.cpp` and `Main.cpp` in lab to your lab TA.