# CSCE 221 Cover Page
# Programming Assignment #1
# Due July 11 by midnight to eCampus

First Name: Suqian Last Name: Wang UIN: 825009505

User Name: wangsuqian123 E-mail address: wangsuqian123@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, read more: Aggie Honor System Office

| Type of sources | Lecture slides | Website | Former notes |
|---|---|---|---|
| People | Teresa Leyk | N/A | Suqian Wang |
| Web pages (provide URL) | ecampus.tamu.edu | www.cplusplus.com | N/A |
| Printed material | N/A | N/A | N/A |
| Other Sources | electronic distribution | | (Electronic)Notes taken from CSCE 121 |

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 Suqian Wang Date July 11, 2017

# Problem Description

This problem has two parts. Part 1 is about implementing a class called My_vec in C++. This class should be able to hold character type data. Part 2 is about implementing a generic version of the class TemplatedMy_vec which can handle different types of data. For both classes, we have three class attributes and several class member functions that can either get information of the object or perform operations on vectors such as remove, insert, replace, sort and so on.

# Data Structures Description

### Theoretical definition

Abstract Data Type (ADT) specifies

- the type of the data stored
- the operations that support the data

The main feature of ADT is

- a clear description of the input to each operation
- the action of each operation
- its return type.

### Real implementation

The type of the data stored

- Part 1: a vector class with char type data
- Part 2: a generic class which can apply to different types of data

The operations that support the data (same for both part 1 and part 2)

constructor, destructor, copy constructor, copy assignment, get_size(), get_capacity(), overloading the [] operator, is_empty(), elem_at_rank(), insert_at_rank(), replace_at_rank(), remove_at_rank(), overloading output operator, find_max_index(), sort_max()

### Analysis of best and worst scenarios for vector

The best scenario for vector

- The vector can successfully perform all kinds of operations without throwing an exception
- The vector can hold different types of data, it is more custom

The worst scenario for vector

- Accessing an element of the vector whose index is out of range

For the time complexity for sorting algorithm, since I used selection sort, so the time complexity for either best case or worse case is the same.

# Instructions to Compile and Run your Program

- Compile: make all
- Run: ./main

## Input and Output Specifications

- Input: there is no terminal input because all test cases were written in main.cpp.
- Output: output is supposed to show the result of each test case such as displaying the vector or displaying the size of the vector after a certain operation. Showing the exception on the screen is also a part of my outputs.

## Logical Exceptions (and bug description)

There are exceptions throughout the program

- Accessing an element whose index is smaller than 0 or greater than size-1, there will be an "Out of Range" exception caught and print on the screen. This exception was considered when overloading [] operator or getting / inserting / replacing / removing an element at index r.
- Operating on an empty vector will cause an exception, which was considered when finding the index of the largest object in an array or sorting the array.

## C++ object oriented or generic programming features, C++11 features

- The My_vec class is a C++ object oriented feature, including the variables of My_vec type defined in main.cpp.
- The Templates used in part 2 is a generic programming feature.
- I didn't use any C++11 feature.

## Testing results

- Part 1

```
[wangsuqian123]@linux2 ~/Wang-Suqian-A1/part1> (21:25:49 07/11/17)
:: ./main
B,
1
A, B,
2
Error: Out of Range
A,
1
E,
1
E,
Error: Out of Range
K,
1
E,
1, 2, 3, 4, 5,
5
index of the largest element in v2 is 4
sorted v2: 1, 2, 3, 4, 5,
Error: Out of Range
```

- Part 2

```
[:: ./main                                                                    ]
***** test objects with char type *****
B
1
A B
2
Error: Out of Range
A
1
E
1
E
Error: Out of Range
K
1
E
1 2 3 4 5
5
index of the largest element in v2 is 4
sorted v2: 1 2 3 4 5
Error: Out of Range

***** test objects with int type *****
5
1
4 5
2
Error: Out of Range
4
1
8
1
8
Error: Out of Range
15
1
8
index of the largest element in vint2 is 0
sorted vint2: 5 13 22 49 90
Error: Out of Range

***** test objects with string type *****
pineapple
1
grape pineapple
2
Error: Out of Range
grape
1
blackberry
1
blackberry
Error: Out of Range
apple
1
blackberry
index of the largest element in vstring2 is 4
sorted vstring2: apple banana orange strawberry watermelon
Error: Out of Range
_
```