CSCE 221 – 505
Lab 1

UIN 822003660
Peng Li

(a) (2 pt) Program Description; Purpose of the Assignment

In this lab, we implemented a vector class that holds char in C++ named My_vec. Next we implemented a generic version of the class My_vec that can handle any type of data. In the class, we created several functions to read the vectors information, edit the elements in the vectors and other operations on the vector.

(b) (4 pt) Data Structures Description

- **Theoretical definition**

Abstract Data Type (ADT) that 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**

In the first part of the lab, we implemented a char type vector class that can do different kind of the operation on the vector. In the second part, we implemented a generic class that can hold different types of data. In both parts, elem_at_rank(), insert_at_rank(), constructor, copy constructor, destructor, assignment operator, overloading the [] operator and, overloading << operator, find_max_index and find_max_index are all works well.

- **Analysis of best and worst scenarios for vector.**

Best part is we can do different operations on the vector and can keep different types of data.

Worst part is we didn't implement concatenation function for the vector class.

(c) (2 pt) Instructions to Compile and Run your Program; Input and Output Specifications

Compile: make all

Run program: ./main

Since we implemented the test program in mian function. There is no terminal input for testing the program.

Output will show the content of the vector and size of the vector after each operation.

(d) (2 pt) Logical Exceptions (And bug description)

When the users try to access on an element that does not exits, the program will automatically exit with a warning "Out of range". Note one exception is that insert an element to rank that is bigger than the current vector size. The function insert_at_rank() will put zeros between the last element and the new added element.

(e) (5 pt) C++ object oriented or generic programming features, C++11 features.

The My_vec class is object oriented programming.

The generic programming is folder part2 file My_vec.h. For part2, we implemented a generic vector that can hold any type of data.

The C++ 11 feature is in folder part1 file My_vec.cpp line 15. I used auto to initialize the variable i.

(f) (5 pt) Testing results

Results for char vector:

```
[abc]@sun
~/CSCE221/lab/Li-Peng-A1/221-14b-A1-
code> (13:45:25 09/12/14)
:: ./main
B
1
A B
2
A B 0 0 0 0 0 0 0 0 D
11
A B 0 0 0 0 0 0 0 0 D
10
A B E 0 0 0 0 0 0 D
10
A B E 0 0 0 0 0 0 D
A B Y 0 0 0 0 0 0 D
K
1
A B Y 0 0 0 0 0 0 D
10
v2 find max is: v2[2]
Y D B A 0 0 0 0 0 0
Out of range
```

Results for generic vector:

```
[abc]@sun
~/CSCE221/lab/Li-Peng-A1/generic>
(13:47:09 09/12/14)
:: ./main
3
1
1 3
2
1 3 0 0 0 0 0 0 0 0 5
11
1 3 0 0 0 0 0 0 0 5
10
1 3 4 0 0 0 0 0 0 5
10
```

```
1 3 4 0 0 0 0 0 0 5
1 3 100 0 0 0 0 0 0 5
50
1
1 3 100 0 0 0 0 0 0 5
10
v2 find max is: v2[2]
100 5 3 1 0 0 0 0 0 0

0
1.444
1
1.444 0 0 0 0 0 0 0 0 0 4.6
11
1.444 0 0 0 0 0 0 0 0 4.6
10
1.444 0 4.5 0 0 0 0 0 0 4.6
10
1.444 0 4.5 0 0 0 0 0 0 4.6
1.444 0 100 0 0 0 0 0 0 4.6
50.44
1
1.444 0 100 0 0 0 0 0 0 4.6
10
v5 find max is: v5[2]
100 4.6 1.444 0 0 0 0 0 0 0
d
1
e d
2
e d       z
11
e d      z
10
e d f      z
10
e d f      z
e d a      z
e
1
e d a      z
```

```
10
v8 find max is: v8[2]
z e d a
```