R6.3

```
    int min, max;
    vector<int> test1;
    test1.push_back(10);
    test1.push_back(5);
    test1.push_back(7);
    test1.push_back(2);
    min=test1[0];
    max=test1[0];
    for(int i=0;i<test1.size()-1;i++)
    {
        if(test1[i]<min)
        {
            min=test1[i];
        }
        if(test1[i]>max)
        {
            max=test1[i];
        }
    }
```

R6.8

```
int num[10];
    for(int i=0;i<10;i++)
    {
        cout << "Enter in a number"<<endl;
        cin >>num[i];
    }
    for(int i=9;i>-1;i--)
    {
        cout<<num[i]<<endl;
    }
```

R6.9

A. If you are sorting a vector then you would do it this way
    a. Void test(vector<int> words);
        i. It would sort the vector and print it out or output it to a file

B. If you were arranging vectors in alphabetical order
    a. Void test(vector<string> &words);
        i. It would print out the words in alphabetical order

C. If you were doing something to the vector and not just printing something out then you would return it
    a. Int test(vector<int> words);
        i. It would return the vector because you are manipulating it

R6.13

```
For(int I=0;i<v.size-1;i++)
{
        if(v[i]>newEmployee)
        {
                v[i].pushback(newEmployee)
        }
}
```

r6.15

```
  a. bool comape(vector<int> &test1, vector<int> &test2)
       a. if(test1==test2)
            {
               return true;
            }
          else
          return false;
  b. void copy(vector<int> &test1, vector<int> &test2)
       a. for(int i=0;i<test1.size()-1;i++)
       b.     {
       c.          test2.push_back(i);
       d.     }
  c. void over(vector<int> &test1)
       a. for(int i=0; i<test1.size()-1;i++)
       b. {
       c.   test1.push_back(i);
       d. }
  d. void erase(vector<int> &test1);
       a. test1.erase (test1.begin(),test1.size()-1);
```