**Written Questions**

1. Pseudo Code
2. Initialize a variable min. Create a int size as the size of the array.
3. Create a recursive function find\_min( T a[], int size). Create a local variable int n inside the function to represent the position of the arry.
4. In the find\_min(T a[], int size) function, use the min as a criterion to compare with the element in the arry. If min is greater than the element, assign the value of the element to min. Else, do nothing change to variables.
5. Use the next element in the array along with (size-1) to implement in the function find\_min(T a[], int size) recursively. For example, return find\_min(T a[n+1],int (size-1))
6. Stop the recursion when the size reaches to 1 and return the min, which is the smallest value in the array.
7. Pseudo Code
8. Create a function double x\_n(double x, int n).
9. If(x==0) if(n=0) return x^n=1 ;if(n>0) return x^n=0.
10. If(n==0) return 1.
11. If (x!=0) return (x\*x\_n(x, n-1)).
12. Selection sort:

Void SelectionSort(vector<int> &v)

{

for( int n=0;n< v.size()-1; n++)

int mid=n;

for( int j=n+1;j<v.size();j++)

if(v[j]<v[mid])

mid=j;

if(mid!=n)

{

swap(v[m],v[mid]);

}

1. Vector of length 0

There will be no change in this situation, since n<0-1, which is -1. The function works correctly when there is zero element in the vector.

1. Vector of length 1

There will be no change in this situation, since n<1-1, which is zero. The function works correctly when there is zero element in the vector.

1. Vector of length 2

Assume that V={1,2};

The n will be determined as zero and j will be determined as one. In this case, v[0]<V[1] and there will make no change for this.

Assume that V={2,1}

The n will be determined as zero and j will be determined as one. In this case, V[0]>V[1]. Firstly, int mid will be assigned the value of j and goes to the if statement to swap these two elements. Finally, we have V={1,2}.

1. Vector of length 3

Assume that V={2,1,3}

Firstly, The first for loop assign n=0 and j=1 and with the help of the if statement, the vector will be switched as V={1,2,3}.

Secondly, n will be assigned as 1 and v[1] is compared with v[n], which are exactly the same. Therefore, the vector has been successfully sorted.

double x=4.5;

double\* a;

a=&x;

double\* b;

b=a;

delete a;

double\* c;

c=NULL;

double\* d;

5.

If you forget to delete an object obtained from the heap, that memory will not be released. Moreover, other programs can use it. This might use up all the free memory.

if you try to delete it a second time, you will get an error message.