Wisely Natalia- 1155113929
CSCI2100B- Assignment 2
Written Exercises

1. Using Selection Sort algorithm to sort sequence in ascending order:

   Initially : **8,** 6, 8, 7, 5, 9, **1**
   1st run : 1, **6,** 8, 7, **5**, 9, 8
   2nd run : 1, 5**, 8**, 7, **6**, 9, 8
   3rd run : 1, 5, 6, **7**, 8, 9, 8
   4th run : 1, 5, 6, 7, **8**, 9, **8**
   5th run : 1, 5, 6, 7, 8, **9**, **8**
   6th run : 1, 5, 6, 7, 8, 8, 9

   Bolded number on the left-hand side is swapped with the right-hand side if its value is greater than or equal to right hand side bolded number. If it is not, it will remain at its position.

2. Computational Complexity = $O(n^2)$

3. (a)

```
int max (listADT L1) {
    if (ListIsEmpty(L1) == 1) {
        exit(EXIT_FAILURE);
    }
    else if (ListIsEmpty(Tail(L1))==1){
        return(Head(L1));
    }
    else {
        int Max = max(Tail(L1));
        if (Max < Head(L1)) {
            return(Head(L1));
        }
        else {
            return(Max);
        }

    }
}
```

3. (b)

```
int lastButOne (listADT L1) {
    if (ListIsEmpty(Tail(Tail(L1)))== 1) {
        return(Head(L1));
    }
    else if (ListIsEmpty(Tail(L1)) == 1) {
        exit(EXIT_FAILURE);
    }
    else {
        return lastButOne (Tail(L1));
    }
}
```

3. (c)

```
int member(int x, listADT L1) {
    if (Head(L1) == x) {
        return(1);
    }
    else if (ListIsEmpty(Tail(L1))==1) {
        return(0);
    }
    else {
        return(member(x,Tail(L1)));
    }
}
```

3. (d)

```
listADT firstN (listADT L1, int x) {
    if (ListIsEmpty(L1)== 1) {
        exit(EXIT_FAILURE);
    }
    else if (x == 1) {
        return(Cons(Head(L1), EmptyList()));
    }
    else {
        x-=1;
        return(Cons(Head(L1),firstN(Tail(L1), x)));
    }
}
```

3 (e)

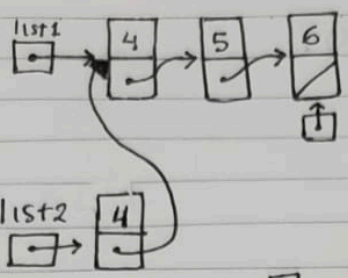
```
listADT afterFour(int x, listADT L1) {
    if(ListIsEmpty(L1)== 1) {
        return(EmptyList());
    }
    else {
        if(Head(L1)== 4){
            return(Cons(4,(Cons(x,Tail(L1)))));
        }
        return(Cons(Head(L1),afterFour(x,Tail(L1))));
    }
}
```

3 (f)

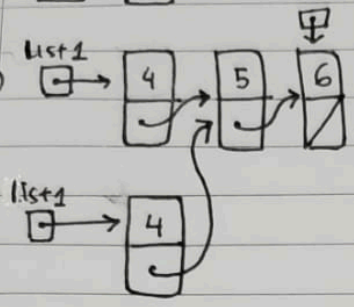
```
int listEqual (listADT L1, listADT L2){
    if (ListIsEmpty(L1)==1 && ListIsEmpty(L2)==1) {
        return(1);
    }
    else if (ListIsEmpty(L1)==1 || ListIsEmpty(L2) ==1) {
        return(0);
    }
    else if (Head(L1)== Head(L2)) {
        return(listEqual(Tail(L1), Tail(L2)));
    }
    return(0);
}
```

DATE :

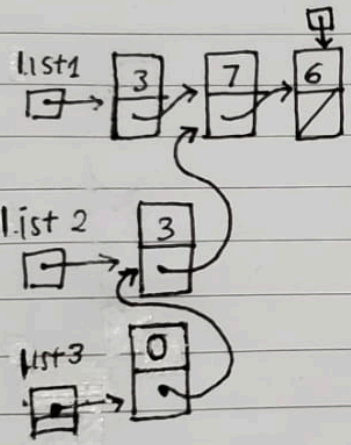
4 (a)



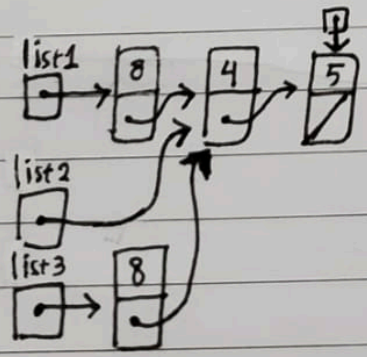
(b)



(c)



(d)



listEqual(list1, list3)

list1 != list3

DATE :

4

(e) listEqual (EmptyList (), EmptyList ())

EmptyList() == EmptyList()

(f)