323.33 Spring 2020 Mock Final - 1 Name:_Andres Quintero

Selection-sort is a sorting method. It can be used to sort integers or strings data, and it can written to sort data in ascending order or in descending order as one desires. Given a list of data file of N items, to sort the data in ascending order, the selection sort works as follows: first, it reads and loads the data into an array; then, the sorting is conducted in iterations. At 1^{st} iteration, it scans from 1^{st} item in the array to the N^{th} item in the array to locates the smallest item, then swap the 1^{st} item of array with the smallest item it found. At 2^{nd} iteration, it scan from 2^{nd} item in the array to the N^{th} item in the array to locates the smallest item, then swap the 2^{nd} item with the smallest item it found in the 2^{nd} iteration. The process repeat to N^{th} iteration when the beginning is at the ending of the array. After sorting, it outputs the sorted array to a output file.

You are to write the specs for selection-sort (in ascending order), mimic the specs you have received for all projects this semester:

1. Programming language (your choice):

C++

2. Input specification. Be precise and brief

inFile (use argv[1]): text file that contains integers separated by a single space

3. Output specification, write whatever you like to see in the out files. Be precise and brief.

outFile (use argv[2]): text file that has an integer per line in ascending order;

3. Write the data structure (use just 1 class). Be precise and brief.

```
selectionSort class
      -Ary (int *)
      - N (int)

      getN(inFile)
      loadAry(inFile)
      selectionSorting()
      printAry(outFile)
```

4. Write the algorithm steps for main (…) and algorithm steps for any methods you may call from main.

```
Step 0: inFile <- given
            outFile <- open

Step 1: getN()
            0: count <- 0
            1: value <- read from inFile
            2: count++
            3: repeat 1-2 until inFile.eof()
            4: N = count
            5: close inFile

Step 2: re-open inFile

Step 3: loadAry(inFile)
            0: initialize Ary to size N
            1: i <- 0
            2: value <- read from inFile
            3: Ary[i] <- value
            4: i++
            5: repeat 2-5 while i < N

Step 4: selectionSorting()
            0: position <- 0;
            1: i <- position+1
            2: if Ary[i] < Ary[position]
                  SWAP them
            3: i++
            4: repeat 2-4 while i < N
            5: position++
            6: repeat 1-6 while position < N
Step 5: printAry(outFile)
            0: i <- 0
            1: output Ary[i] -> outFile
            2: i++
            3: repeat 1-3 while i < N

Step 6: close all files
```

5. Follow your specs to implement your Selection sort, in ascending order.

III. Submission: include the following in the same email:

- The essay questions and answers in any option you choose in the above.
- soft copy of programming question
- hard copy :