Sorting of list items – Selection Sorting

- **SELECTION SORT** is a comparison sorting algorithm that is used to sort a random list of items in ascending order.
- The comparison does not require a lot of extra space. It only requires one extra memory space for the temporal variable.
- This is known as **in-place** sorting.
- The list is divided into two partitions:
    a) The first list contains sorted items,
    b) while the second list contains unsorted items.
- By default, the sorted list is empty, and the unsorted list contains all the elements.
- The unsorted list is then scanned for the minimum value, which is then placed in the sorted list.
- This process is repeated until all the values have been compared and sorted.

Visual Representation

Given a list of five elements, the following images illustrate how the selection sort algorithm iterates through the values when sorting them.

The following image shows the unsorted list

Step 1)

The first value 21 is compared with the rest of the values to check if it is the minimum value.

3 is the minimum value, so the positions of 21 and 3 are swapped. The values with a green background represent the sorted partition of the list.

Step 2)

The value 6 which is the first element in the unsorted partition is compared with the rest of the values to find out if a lower value exists

The value 6 is the minimum value, so it maintains its position.

Step 3)

The first element of the unsorted list with the value of 9 is compared with the rest of the values to check if it is the minimum value.

The value 9 is the minimum value, so it maintains its position in the sorted partition.

Step 4)

The value 33 is compared with the rest of the values.

The value 21 is lower than 33, so the positions are swapped to produce the above new list.

Step 5)

We only have one value left in the unpartitioned list. Therefore, it is already sorted.

The final list is like the one shown in the above image.

Example: Python program for implementation of Selection Sort

```
def selectionSort(itemsList ):

    n = len(itemsList)
    for i in range(n - 1):
        minValueIndex = i

        for j in range( i + 1, n):
            if itemsList[j] < itemsList[minValueIndex] :
                minValueIndex = j

        if minValueIndex != i :
            temp = itemsList[i]
            itemsList[i] = itemsList[minValueIndex]
            itemsList[minValueIndex] = temp

    return itemsList


list_numbers = [100, 25, 0, -1, 21,88, 16,99,133,23]
print("\nList before Sorting")
print(list_numbers)

print("\nList after applying Selection Sort technique:")
print(selectionSort(list_numbers))
```

```
(base) F:\CSE1001\Python-Programs>python prgm25-SelectionSorting.py

List before Sorting
[100, 25, 0, -1, 21, 88, 16, 99, 133, 23]

List after applying Selection Sort technique:
[-1, 0, 16, 21, 23, 25, 88, 99, 100, 133]
```

Activities

1) Write an algorithm for selection sorting technique.
2) Draw a flow chart for selection sorting technique.
3) Write a python program for the following requirements.
    a) Read **6** numbers from users and insert them into a list (say **list_numbers**). Make sure that **list_numbers** contains negative, positive, odd, even numbers.
    b) Print the **list_numbers**.
    c) Using **selection sorting technique**, sort the numbers in **list_numbers** and print them in ascending order. Also store the numbers in ascending order into a file (say 'ascending_numbers.txt').
    d) Store the numbers in descending order into a file (say 'descending_numbers.txt').
    e) Take the even numbers from **list_numbers** and store them into a file (say 'even_numbers.txt').
    f) Take the odd numbers from **list_numbers** and store them into a file (say 'odd_numbers.txt').
    g) Trace the entire program manually (handwritten material) in neat format to show the status of all variables and condition checking during all iterations. Add the scanned soft copy into the answer sheet (PDF file).
4) Write down the answers for the following.
    a. Advantages and disadvantages of bubble sorting technique.
    b. Advantages and disadvantages of selection sorting technique.
    c. Difference between bubble and selection sorting techniques.