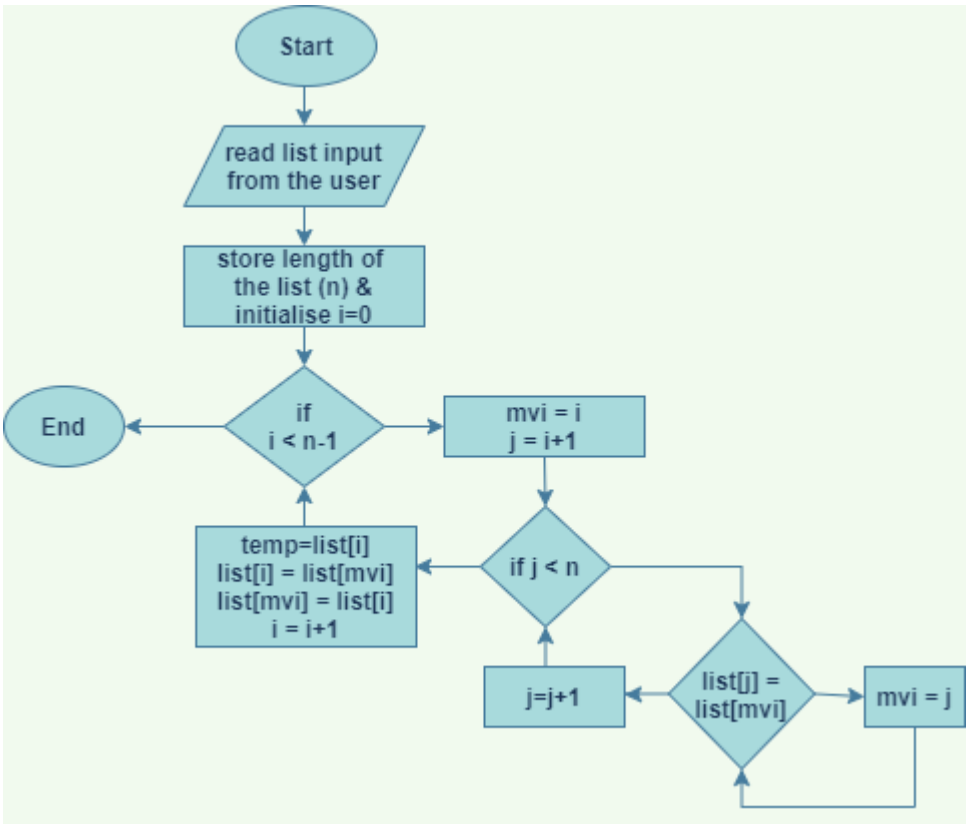
AKSHAT KUMAR
20MIS0183

Activity 1.32

algorithm for selection sorting

1. Start
2. Read an unsorted list from the user and store its length.
3. The outer "for loop" takes each value from the list
4. The value of outer "for loop" is assigned as Mvi
5. Inner "for loop" runs from a value greater than the "outer for loop"
6. It is compared with the Mvi
if it is less than Mvi
the value of Mvi is swapped (the value of j is incremented upto its range)
7. If the Mvi is not equal to the "outer for loop"
Mvi is swapped with "outer for loop" value
8. This loop runs until it is terminated & the list is sorted.
9. End

flowchart for selection sorting



AKSHAT KUMAR
20MIS0183

practice code

```
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))
```

output

```
C:\Users\user\Desktop\python>C:/Users/user/AppData/Local/Programs/Python/Python38-
32/python.exe c:/Users/user/Desktop/python/act1.32/pcc.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]

C:\Users\user\Desktop\python>
```

AKSHAT KUMAR
20MIS0183

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).

```
def selection_sort(given_list):
    length = len(given_list)
    for x in range(length-1):
        minimum_value_holder = x
        for j in range(x + 1, length):
            if given_list[j] < given_list[minimum_value_holder]:
                minimum_value_holder = j
        if minimum_value_holder != x:
            temp = given_list[x]
            given_list[x] = given_list[minimum_value_holder]
            given_list[minimum_value_holder] = temp


def selection_sort_dec(given_list):
    length = len(given_list)
    for x in range(length-1):
        minimum_value_holder = x
        for j in range(x + 1, length):
            if given_list[j] > given_list[minimum_value_holder]:
                minimum_value_holder = j
        if minimum_value_holder != x:
            temp = given_list[x]
            given_list[x] = given_list[minimum_value_holder]
            given_list[minimum_value_holder] = temp



list_numbers = []
```

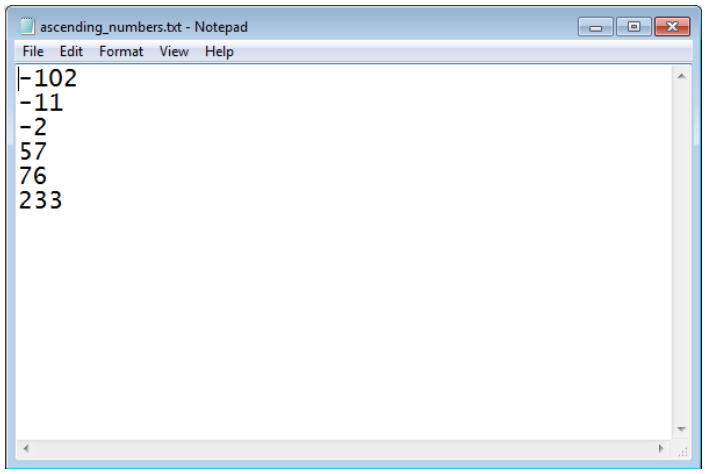
```
print("Enter 6 numbers")
i = 1
while i <= 6:
    number = int(input("Number {} :".format(i)))
    if number not in list_numbers:
        list_numbers.append(number)
    else:
        print("You have already entered this Number, Try another")
        continue
    i += 1

print('List created :', list_numbers, '\n')
# ------------------------------------------------------------------------------------
-------
print('List is sorted in ascending order')
selection_sort(list_numbers)

print(list_numbers)
file_handler = open('ascending_numbers.txt', 'a')
for t in list_numbers:
    file_handler.write(str(t))
    file_handler.write('\n')

print('Created a file ascending_numbers\n')
file_handler.close()
# ------------------------------------------------------------------------------------
---------
print('List is sorted in descending order')
selection_sort_dec(list_numbers)
print(list_numbers)
file_handler = open('descending_numbers.txt', 'a')
for t in list_numbers:
    file_handler.write(str(t))
    file_handler.write('\n')

print('Created a file descending_numbers\n')
file_handler.close()
# ------------------------------------------------------------------------------------
---------
file_handler_even = open('even_numbers.txt', 'a')
file_handler_odd = open('odd_numbers.txt', 'a')
for z in list_numbers:
```

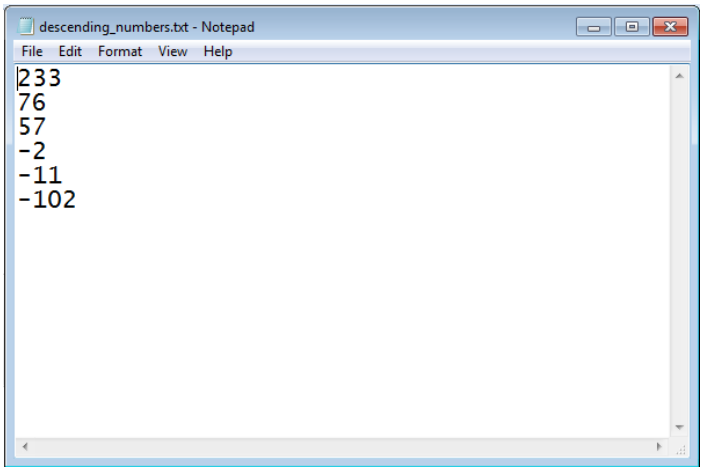
```
    if z % 2 == 0:
        file_handler_even.write(str(z))
        file_handler_even.write('\n')
    else:
        file_handler_odd.write(str(z))
        file_handler_odd.write('\n')

print('Created a file even_numbers\n')
print('Created a file odd_numbers\n')

file_handler_odd.close()
file_handler_even.close()
```

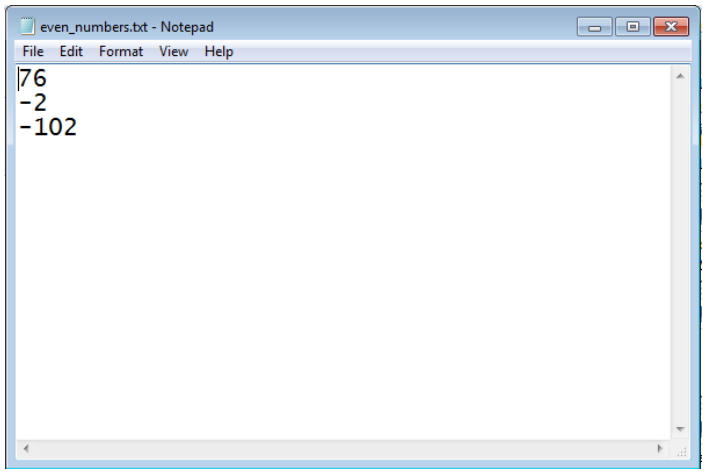
#output

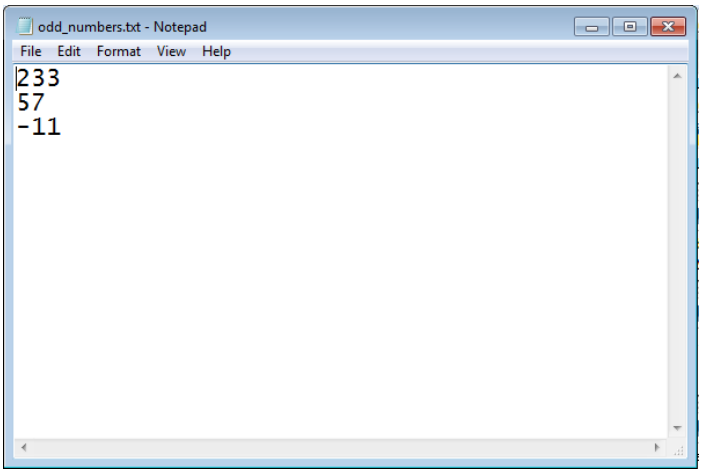
```
C:\Users\user\Desktop\python>C:/Users/user/AppData/Local/Programs/Python/Python38-32/python.exe c:/Users/user/Desktop/python/act1.32/1.py

Enter 6 numbers
Number 1 :76
Number 2 :-2
Number 3 :233
Number 4 :-102
Number 5 :57
Number 6 :-11
List created : [76, -2, 233, -102, 57, -11]

List is sorted in ascending order
[-102, -11, -2, 57, 76, 233]
Created a file ascending_numbers

List is sorted in descending order
[233, 76, 57, -2, -11, -102]
Created a file descending_numbers

Created a file even_numbers

Created a file odd_numbers
```

Name	Date modified	Type	Size
act1.32	1/10/2021 12:10 PM	File folder	
ascending_numbers.txt	1/10/2021 12:08 PM	Text Document	1 KB
descending_numbers.txt	1/10/2021 12:08 PM	Text Document	1 KB
even_numbers.txt	1/10/2021 12:08 PM	Text Document	1 KB
odd_numbers.txt	1/10/2021 12:08 PM	Text Document	1 KB









list = [2, 0, 100, 8, -2]
index: 0 1 2 3 4

index[0] = 2, index[1] = 0, index[2] = 100, index[3] = 8, index[4] = -2

length = 5

for i in range(0,4):
(outer for loop 0,1,2,3)

mVI = 0

for j in range(1,5): [range 1,2,3,4]

j = 1 (1st iteration)
i = 0
if Ls[j] < L[mVI]
if [1] < [0]
if 0 < 2 : true
mVI = j = 1 (swapping takes place)

---

j = 2 (2nd iteration)
i = 0
if Ls[j] < L[mVI]
if [2] < [0]
If 100 < 2 : false
condition is wrong (no swapping)

↳ no change in mVI

---

j = 3 (3rd iteration)
i = 0
if Ls[j] < L[mVI]
if Ls[3] < L[1]
8 < 0 : false
condition is wrong ; (no swapping)

↳ no change in mVI

---

j = 4
i = 0
if Ls[j] < L[mVI]
if Ls[4] < L[1]
-2 < 0 : true
mVI = j = 4 (swapping takes place)

Next Inner loop:

if mvI != i:

4 ≠ 0

So index[0] ⇌ index[4]

After 1st iteration of outer for loop the list will be following — [-2, 0, 100, 8, 2]
0 1 2 3 4

Second iteration of outer for loop

now i = 1

inner for loop

for j in (2, 4)

if [2] < iL [1]

if 100 < 1 ; false

no change in mvI (no swapping)

mvI = 1

---

j = 3

if L[3] < L[1]

if 8 < 0 ; false

no change in mvI (no swapping)

mvI = 1

---

j = 4

if L[4] < L[1]

if 2 < 0 ; false

no change in mvI

mvI = 1 (no swapping)

---

third iteration

now i = 2

j = 3, 4

mvI = 2

if L[j] < L[mvI]

if L[3] < L[2]

if 8 < 100 ; true

mvI = j = 3 (swapping takes place)

---

j = 4
if L[4] < L[3]
if 2 < 8 ; true
mVI = 4 = j (swapping takes place)

inner

mVI ≠ i
4 ≠ 2
[4] ⇌ [2]
2 ⇌ 100

After third iteration of outer for loop
list will be [-2, 0, 2, 8, 100]
0 1 2 3 4

Fourth iteration

i = 3
mVI = 3
j = 4
if L[j] < L[mVI]
if L[4] < L[3]
100 < 8 ; False
mVI = 3 (no swapping)
j = 5 (terminated)
inner loop
if 3 ≠ 3 false

Fifth iteration

i = 4
loop will be terminated
return (itemList)

List After Sorting → [-2, 0, 2, 8, 100]
0 1 2 3 4