GROUP MEMBERS:

- SYED AZEEM UL MAZHAR (EP1849107)
- TAHA AHMED KHAN (EP1849120)
- TANIS ANIS (EP1849123)
- ZAINAB ASIF (EP1849134)

CONCEPTS OF OPERATING SYSTEMS (BSCS – 502)

SCHEDULING ALGORITHM ASSIGNMENT

SHORTEST JOB FIRST & SHORTEST REMAINING TIME

SUBMITTED TO:

SIR ASHER ALI

SHORTEST JOB FIRST

```
import random
import numpy
from numpy.random import randint
new_list = []
list_len = []
unsort_list = []
for pro in range (5):
    process = []
    ran_len = random.randint(2,9)
    list_len.append(ran_len)
    for j in range (ran_len):
        ran_num = random.randint(0,20)
        process.append(ran_num)
    new_list.append(process)
for i in new_list:
    print("process:", i)
    unsort_list.append(len(i))
sort_list = numpy.sort(unsort_list)
print("-----------------------------------------")
print("unsorted length list:", unsort_list)
print("-----------------------------------------")
print("sorted length list:", sort_list)
print("-----------------------------------------")
for j in sort_list:
    for SJF in new_list:
        if (j==len(SJF)):
            new_list.remove(SJF)
            print("process after SJF:", SJF)
            break
print("\n--- ALL PROCESSES COMPLETED ---")
```

OUTPUT:

```
process: [6, 9, 13, 13, 11, 11, 6, 8, 16]
process: [2, 17]
process: [12, 20, 6, 7, 2, 20, 10]
process: [4, 8]
process: [5, 1, 12, 9, 10, 15, 18, 2]
-----------------------------------------
unsorted length list: [9, 2, 7, 2, 8]
-----------------------------------------
sorted length list: [2 2 7 8 9]
-----------------------------------------
process after SJF: [2, 17]
process after SJF: [4, 8]
process after SJF: [12, 20, 6, 7, 2, 20, 10]
process after SJF: [5, 1, 12, 9, 10, 15, 18, 2]
process after SJF: [6, 9, 13, 13, 11, 11, 6, 8, 16]

--- ALL PROCESSES COMPLETED ---
>
```

SHORTEST REMAINING TIME

```
import random
arr_list = [ [] , [] , [] , [] , [] ]
for i in range(5):
    for a in range(random.randint(3,8)):
        b = random.randint(1,15)
        arr_list[i].append(b)

pro1 = arr_list[0]
print("length =", len(pro1), "=>", "process 1:", pro1)
pro2 = arr_list[1]
print("length =", len(pro2), "=>", "process 2:", pro2)
pro3 = arr_list[2]
print("length =", len(pro3), "=>", "process 3:", pro3)
pro4 = arr_list[3]
print("length =", len(pro4), "=>", "process 4:", pro4)
pro5 = arr_list[4]
print("length =", len(pro5), "=>", "process 5:", pro5)
print("----------------------------------------")

print("process 1:", pro1, "\nthread executed:", pro1[0])
del pro1[0]
print("----------------------------------------")

print("process 1:", pro1, "\nprocess 2:", pro2)
if len(pro1) > len(pro2):
    print("thread executed:", pro2[0])
    del pro2[0]
else:
    print("thread executed:" ,pro1[0])
    del pro1[0]
```

```
print("-----------------------------------------")

print("process 1:", pro1, "\nprocess 2:", pro2,
"\nprocess 3:", pro3)
if len(pro1) <= len(pro2) and len(pro3):
    print("thread executed:", pro1[0])
    del pro1[0]
elif len(pro2) <= len(pro1) and len(pro3):
    print("thread executed:", pro2[0])
    del pro2[0]
elif len(pro3) <= len(pro1) and len(pro2):
    print("thread executed:", pro3[0])
    del pro3[0]
print("-----------------------------------------")

print("process 1:", pro1, "\nprocess 2:", pro2,
"\nprocess 3:", pro3, "\nprocess 4:", pro4)
if len(pro1) <= len(pro2) and len(pro3) and len(pro4):
    print("thread executed:", pro1[0])
    del pro1[0]
elif len(pro2) <= len(pro1) and len(pro3) and len(pro4):
    print("thread executed:", pro2[0])
    del pro2[0]
elif len(pro3) <= len(pro1) and len(pro2) and len(pro4):
    print("thread executed:", pro3[0])
    del pro3[0]
elif len(pro4) <= len(pro1) and len(pro2) and len(pro3):
    print("thread executed:", pro4[0])
    del pro4[0]
print("-----------------------------------------")
```

```
print("process 1:", pro1, "\nprocess 2:", pro2,
"\nprocess 3:", pro3, "\nprocess 4:", pro4, "\nprocess
5:", pro5)

while len(pro1)>0 or len(pro2)>0 or len(pro3)>0 or
len(pro4)>0 or len(pro5)>0:

    len_list = [len(pro1), len(pro2), len(pro3),
len(pro4), len(pro5)]

    m = min(i for i in len_list if i > 0)

    if len(pro1) > 0 and m == len(pro1):
        print("thread executed:", pro1[0])
        del pro1[0]
    elif len(pro2) > 0 and m == len(pro2):
        print("thread executed:", pro2[0])
        del pro2[0]
    elif len(pro3) > 0 and m == len(pro3):
        print("thread executed:", pro3[0])
        del pro3[0]
    elif len(pro4) > 0 and m == len(pro4):
        print("thread executed:", pro4[0])
        del pro4[0]
    elif len (pro5) > 0 and m == len(pro5):
        print("thread executed:", pro5[0])
        del pro5[0]
    print("----------------------------------------")
    print("process 1:", pro1, "\nprocess 2:", pro2,
"\nprocess 3:", pro3, "\nprocess 4:", pro4, "\nprocess
5:", pro5)
print("\n--- ALL PROCESSES COMPLETED ---")
```

OUTPUT:

```
length = 5 => process 1: [7, 11, 2, 10, 2]
length = 7 => process 2: [7, 6, 11, 11, 15, 15, 8]
length = 5 => process 3: [12, 12, 4, 7, 5]
length = 3 => process 4: [2, 12, 5]
length = 8 => process 5: [12, 14, 5, 3, 11, 2, 11, 14]
----------------------------------------
process 1: [7, 11, 2, 10, 2]
thread executed: 7
----------------------------------------
process 1: [11, 2, 10, 2]
process 2: [7, 6, 11, 11, 15, 15, 8]
thread executed: 11
----------------------------------------
process 1: [2, 10, 2]
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
thread executed: 2
----------------------------------------
process 1: [10, 2]
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
process 4: [2, 12, 5]
thread executed: 10
----------------------------------------
process 1: [2]
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
process 4: [2, 12, 5]
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 2
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
```

```
process 4: [2, 12, 5]
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 2
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
process 4: [12, 5]
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 12
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
process 4: [5]
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 5
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 12, 4, 7, 5]
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 12
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [12, 4, 7, 5]
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 12
----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [4, 7, 5]
process 4: []
```

```
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 4
-----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [7, 5]
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 7
-----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: [5]
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 5
-----------------------------------------
process 1: []
process 2: [7, 6, 11, 11, 15, 15, 8]
process 3: []
process 4:[]
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 7
-----------------------------------------
process 1: []
process 2: [6, 11, 11, 15, 15, 8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 6
-----------------------------------------
process 1: []
process 2: [11, 11, 15, 15, 8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
```

```
thread executed: 11
----------------------------------------
process 1: []
process 2: [11, 15, 15, 8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 11
----------------------------------------
process 1: []
process 2: [15, 15, 8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 15
----------------------------------------
process 1: []
process 2: [15, 8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 15
----------------------------------------
process 1: []
process 2: [8]
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 8
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [12, 14, 5, 3, 11, 2, 11, 14]
thread executed: 12
```

```
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [14, 5, 3, 11, 2, 11, 14]
thread executed: 14
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [5, 3, 11, 2, 11, 14]
thread executed: 5
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [3, 11, 2, 11, 14]
thread executed: 3
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [11, 2, 11, 14]
thread executed: 11
----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [2, 11, 14]
thread executed: 2
----------------------------------------
```

```
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [11, 14]
thread executed: 11
-----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: [14]
thread executed: 14
-----------------------------------------
process 1: []
process 2: []
process 3: []
process 4: []
process 5: []

--- ALL PROCESSES COMPLETED ---
```

--- END ---