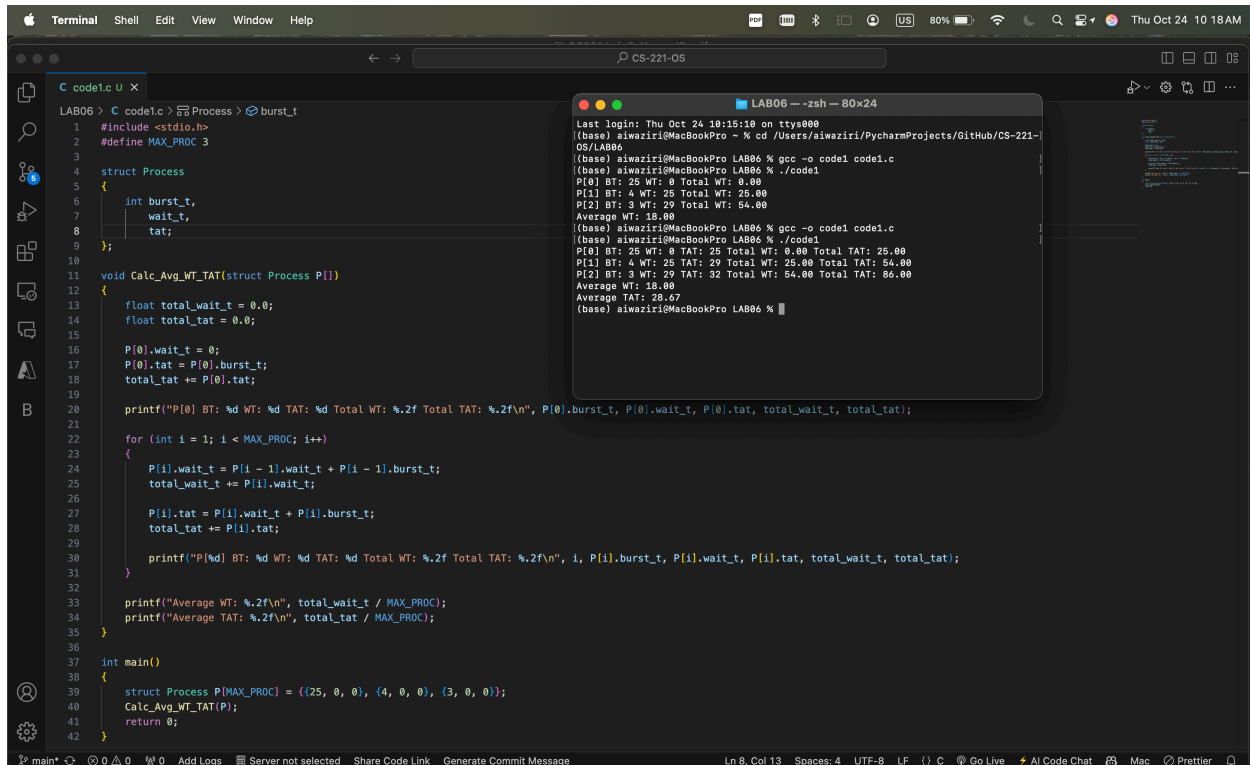


Name: Abubakar Waziri

ID: 4220056

LAB06

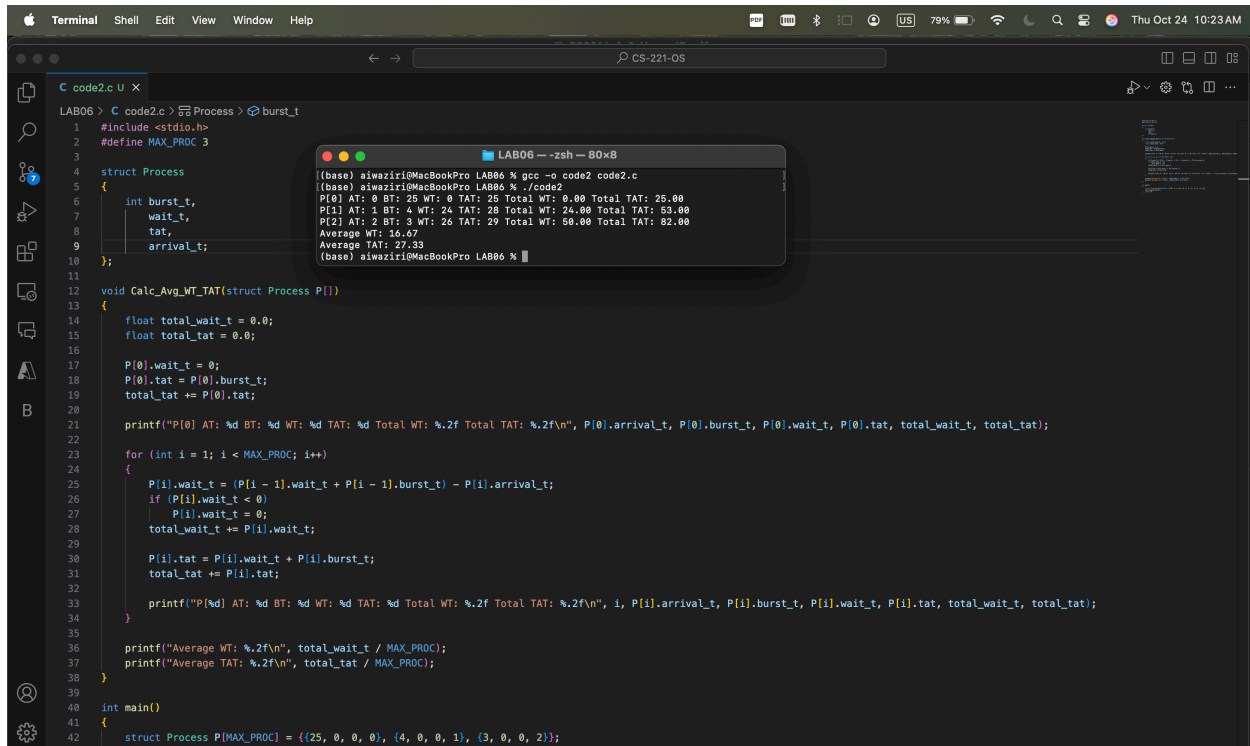
Q1:



```
LAB06 > C code1.c U X
LAB06 > C code1.c > Process > burst_t
1 #include <stdio.h>
2 #define MAX_PROCS 3
3
4 struct Process
5 {
6     int burst_t,
7     wait_t,
8     tat;
9 };
10
11 void Calc_Avg_WT_TAT(struct Process P[])
12 {
13     float total_wait_t = 0.0;
14     float total_tat = 0.0;
15
16     P[0].wait_t = 0;
17     P[0].tat = P[0].burst_t;
18     total_tat += P[0].tat;
19
20     printf("P[0] BT: %d WT: %d TAT: %d Total WT: %.2f Total TAT: %.2f\n", P[0].burst_t, P[0].wait_t, P[0].tat, total_wait_t, total_tat);
21
22     for (int i = 1; i < MAX_PROCS; i++)
23     {
24         P[i].wait_t = P[i - 1].wait_t + P[i - 1].burst_t;
25         total_wait_t += P[i].wait_t;
26
27         P[i].tat = P[i].wait_t + P[i].burst_t;
28         total_tat += P[i].tat;
29
30         printf("P[%d] BT: %d WT: %d TAT: %d Total WT: %.2f Total TAT: %.2f\n", i, P[i].burst_t, P[i].wait_t, P[i].tat, total_wait_t, total_tat);
31     }
32
33     printf("Average WT: %.2f\n", total_wait_t / MAX_PROCS);
34     printf("Average TAT: %.2f\n", total_tat / MAX_PROCS);
35 }
36
37 int main()
38 {
39     struct Process P[MAX_PROCS] = {{25, 0, 0}, {4, 0, 0}, {3, 0, 0}};
40     Calc_Avg_WT_TAT(P);
41     return 0;
42 }
```

```
Last login: Thu Oct 24 18:15:10 on ttys000
(base) aiwaziri@MacBookPro ~ % cd /Users/aiwaziri/PycharmProjects/GitHub/CS-221-OS/LAB06
(base) aiwaziri@MacBookPro LAB06 % gcc -o code1 code1.c
(base) aiwaziri@MacBookPro LAB06 % ./code1
P[0] BT: 25 WT: 0 Total WT: 0.00
P[1] BT: 4 WT: 25 Total WT: 25.00
P[2] BT: 3 WT: 29 Total WT: 54.00
Average WT: 18.00
(base) aiwaziri@MacBookPro LAB06 % gcc -o code1 code1.c
(base) aiwaziri@MacBookPro LAB06 % ./code1
P[0] BT: 25 WT: 0 TAT: 25 Total WT: 0.00 Total TAT: 25.00
P[1] BT: 4 WT: 25 TAT: 29 Total WT: 25.00 Total TAT: 54.00
P[2] BT: 3 WT: 29 TAT: 32 Total WT: 54.00 Total TAT: 86.00
Average WT: 18.00
Average TAT: 28.67
(base) aiwaziri@MacBookPro LAB06 %
```

Q2:



```
LAB06 > C code2.c > Process > burst_t
1 #include <stdio.h>
2 #define MAX_PROC 3
3
4 struct Process
5 {
6     int burst_t,
7     wait_t,
8     tat,
9     arrival_t;
10 };
11
12 void Calc_Avg_WT_TAT(struct Process P[])
13 {
14     float total_wait_t = 0.0;
15     float total_tat = 0.0;
16
17     P[0].wait_t = 0;
18     P[0].tat = P[0].burst_t;
19     total_tat += P[0].tat;
20
21     printf("P[0] AT: %d BT: %d WT: %d TAT: %d Total WT: %.2f Total TAT: %.2f\n", P[0].arrival_t, P[0].burst_t, P[0].wait_t, P[0].tat, total_wait_t, total_tat);
22
23     for (int i = 1; i < MAX_PROC; i++)
24     {
25         P[i].wait_t = (P[i - 1].wait_t + P[i - 1].burst_t) - P[i].arrival_t;
26         if (P[i].wait_t < 0)
27             P[i].wait_t = 0;
28         total_wait_t += P[i].wait_t;
29
30         P[i].tat = P[i].wait_t + P[i].burst_t;
31         total_tat += P[i].tat;
32
33         printf("P[%d] AT: %d BT: %d WT: %d TAT: %d Total WT: %.2f Total TAT: %.2f\n", i, P[i].arrival_t, P[i].burst_t, P[i].wait_t, P[i].tat, total_wait_t, total_tat);
34     }
35
36     printf("Average WT: %.2f\n", total_wait_t / MAX_PROC);
37     printf("Average TAT: %.2f\n", total_tat / MAX_PROC);
38 }
39
40 int main()
41 {
42     struct Process P[MAX_PROC] = {{25, 0, 0, 0}, {4, 0, 0, 1}, {3, 0, 0, 2}};
```

```
LAB06 -- -zsh -- 80x8
(base) aiwaziri@MacBookPro LAB06 % gcc -o code2 code2.c
(base) aiwaziri@MacBookPro LAB06 % ./code2
P[0] AT: 0 BT: 25 WT: 0 TAT: 25 Total WT: 0.00 Total TAT: 25.00
P[1] AT: 1 BT: 4 WT: 24 TAT: 28 Total WT: 24.00 Total TAT: 53.00
P[2] AT: 2 BT: 3 WT: 26 TAT: 29 Total WT: 50.00 Total TAT: 82.00
Average WT: 16.67
Average TAT: 27.33
(base) aiwaziri@MacBookPro LAB06 %
```

Q3:

The screenshot shows a macOS Terminal window with a code editor. The code is in C and implements a process scheduling algorithm. It defines a `Process` struct with fields `pid`, `burst_t`, `wait_t`, and `tat`. It includes functions for sorting processes by burst time and calculating average wait time and turnaround time (TAT).

```
LAB06 > C sjfc > Process > pid
1 #include <stdio.h>
2
3 #define MAX_PROCS 5
4
5 struct Process
6 {
7     int pid;
8     int burst_t;
9     int wait_t;
10    int tat;
11 };
12
13 void sort_by_burst_time(struct Process P[], int n)
14 {
15     for (int i = 0; i < n - 1; i++)
16     {
17         for (int j = 0; j < n - i - 1; j++)
18         {
19             if (P[j].burst_t > P[j + 1].burst_t)
20             {
21                 struct Process temp = P[j];
22                 P[j] = P[j + 1];
23                 P[j + 1] = temp;
24             }
25         }
26     }
27 }
28
29 void Calc_Avg_WT_TAT(struct Process P[], int n)
30 {
31     float total_wait_t = 0.0;
32     float total_tat = 0.0;
33
34     P[0].wait_t = 0;
35     P[0].tat = P[0].burst_t;
36     total_tat += P[0].tat;
37
38     for (int i = 1; i < n; i++)
39     {
40         P[i].wait_t = P[i - 1].wait_t + P[i - 1].burst_t;
41         total_wait_t += P[i].wait_t;
42     }
43
44     P[i].tat = P[i].wait_t + P[i].burst_t;
45 }
```

The output of the program is shown in a separate window titled "LAB06 -- zsh -- 93x11":

```
((base) aiwaziri@MacBookPro LAB06 % gcc -o sjf sjf.c
((base) aiwaziri@MacBookPro LAB06 % ./sjf
PID    BT    WT    TAT
4      3     0     3
5      4     3     7
1      6     7    13
3      7    13    20
2      8    20    28
Average WT: 8.60
Average TAT: 14.20
(base) aiwaziri@MacBookPro LAB06 %
```