BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

HỆ ĐIỀU HÀNH – THỰC HÀNH

BÁO CÁO LAB04

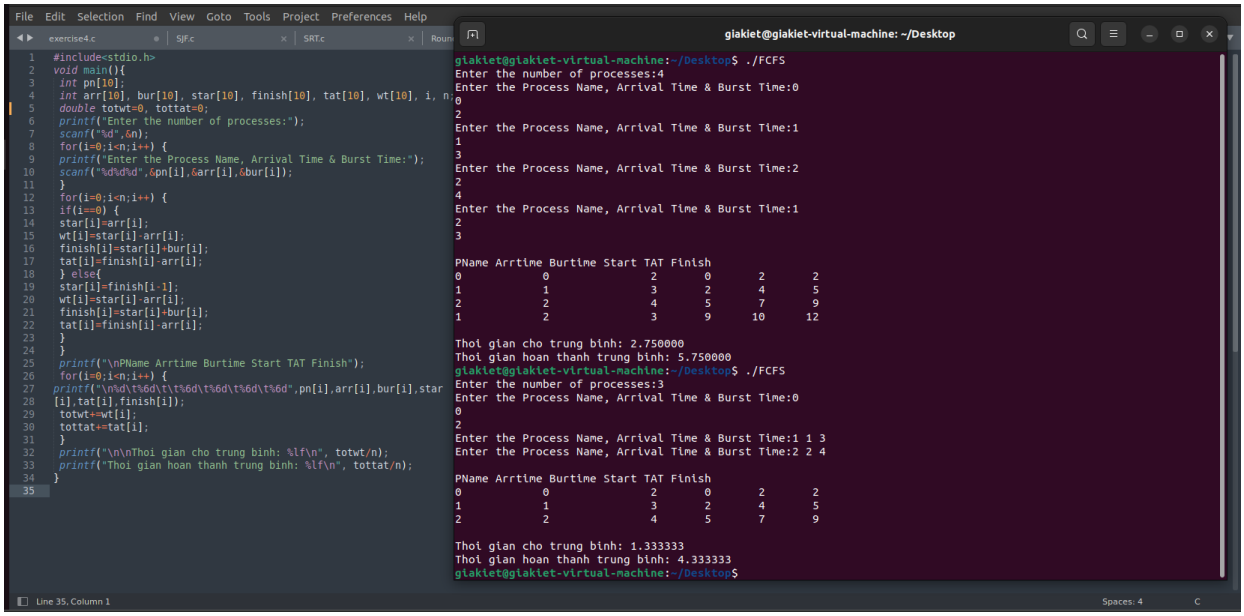
Lớp: IT007.N12.KHCL

Tên: Lê Gia Kiệt

MSSV: 21522255

BÀI LÀM

- Soạn thảo và biên dịch giải thuật FCFS bên dưới
- Bổ sung code để tính average waiting time và average turnaround.



```
#include<stdio.h>
void main(){
 int pn[10];
 int arr[10], bur[10], star[10], finish[10], tat[10], wt[10], i, n;
 double totwt=0, tottat=0;
 printf("Enter the number of processes:");
 scanf("%d",&n);
 for(i=0;i<n;i++) {
 printf("Enter the Process Name, Arrival Time & Burst Time:");
 scanf("%d%d%d",&pn[i],&arr[i],&bur[i]);
 }
 for(i=0;i<n;i++) {
 if(i==0) {
 star[i]=arr[i];
 wt[i]=star[i]-arr[i];
 finish[i]=star[i]+bur[i];
 tat[i]=finish[i]-arr[i];
 } else{
 star[i]=finish[i-1];
 wt[i]=star[i]-arr[i];
 finish[i]=star[i]+bur[i];
 tat[i]=finish[i]-arr[i];
 }
 }
 printf("\nPName Arrtime Burtime Start TAT Finish");
 for(i=0;i<n;i++) {
printf("\n%d\t%6d\t\t%6d\t%6d\t%6d\t%6d",pn[i],arr[i],bur[i],star
[i],tat[i],finish[i]);
 totwt+=wt[i];
 tottat+=tat[i];
 }
 printf("\n\nThoi gian cho trung binh: %lf\n", totwt/n);
 printf("Thoi gian hoan thanh trung binh: %lf\n", tottat/n);
}
```

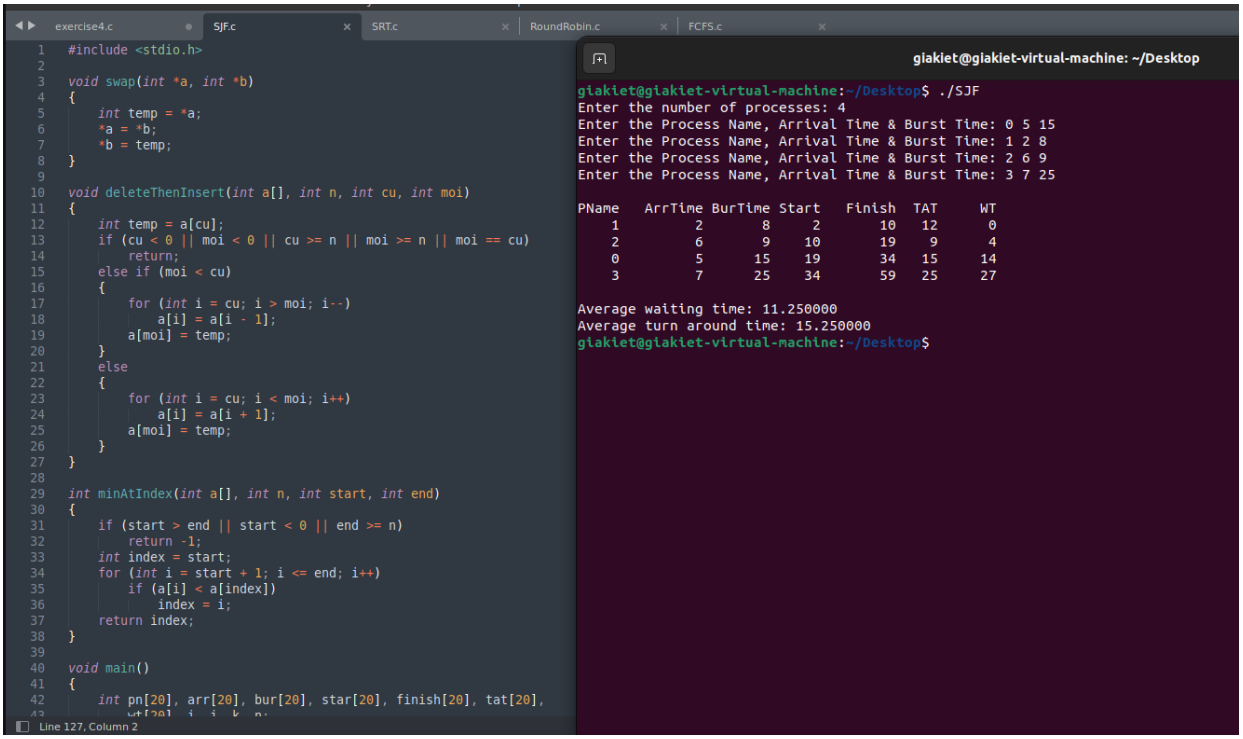
```
giakiet@giakiet-virtual-machine:~/Desktop$ ./FCFS
Enter the number of processes:4
Enter the Process Name, Arrival Time & Burst Time:0
0
2
Enter the Process Name, Arrival Time & Burst Time:1
1
3
Enter the Process Name, Arrival Time & Burst Time:2
2
4
Enter the Process Name, Arrival Time & Burst Time:1
2
3

PName Arrtime Burtime Start TAT Finish
0           0           2       0       2       2
1           1           3       2       4       5
2           2           4       5       7       9
1           2           3       9       10      12

Thoi gian cho trung binh: 2.750000
Thoi gian hoan thanh trung binh: 5.750000
giakiet@giakiet-virtual-machine:~/Desktop$ ./FCFS
Enter the number of processes:3
Enter the Process Name, Arrival Time & Burst Time:0
0
2
Enter the Process Name, Arrival Time & Burst Time:1 1 3
Enter the Process Name, Arrival Time & Burst Time:2 2 4

PName Arrtime Burtime Start TAT Finish
0           0           2       0       2       2
1           1           3       2       4       5
2           2           4       5       7       9

Thoi gian cho trung binh: 1.333333
Thoi gian hoan thanh trung binh: 4.333333
giakiet@giakiet-virtual-machine:~/Desktop$
```

1. Viết chương trình mô phỏng giải thuật SJF với các yêu cầu sau:

   ❖ Nhập số lượng process

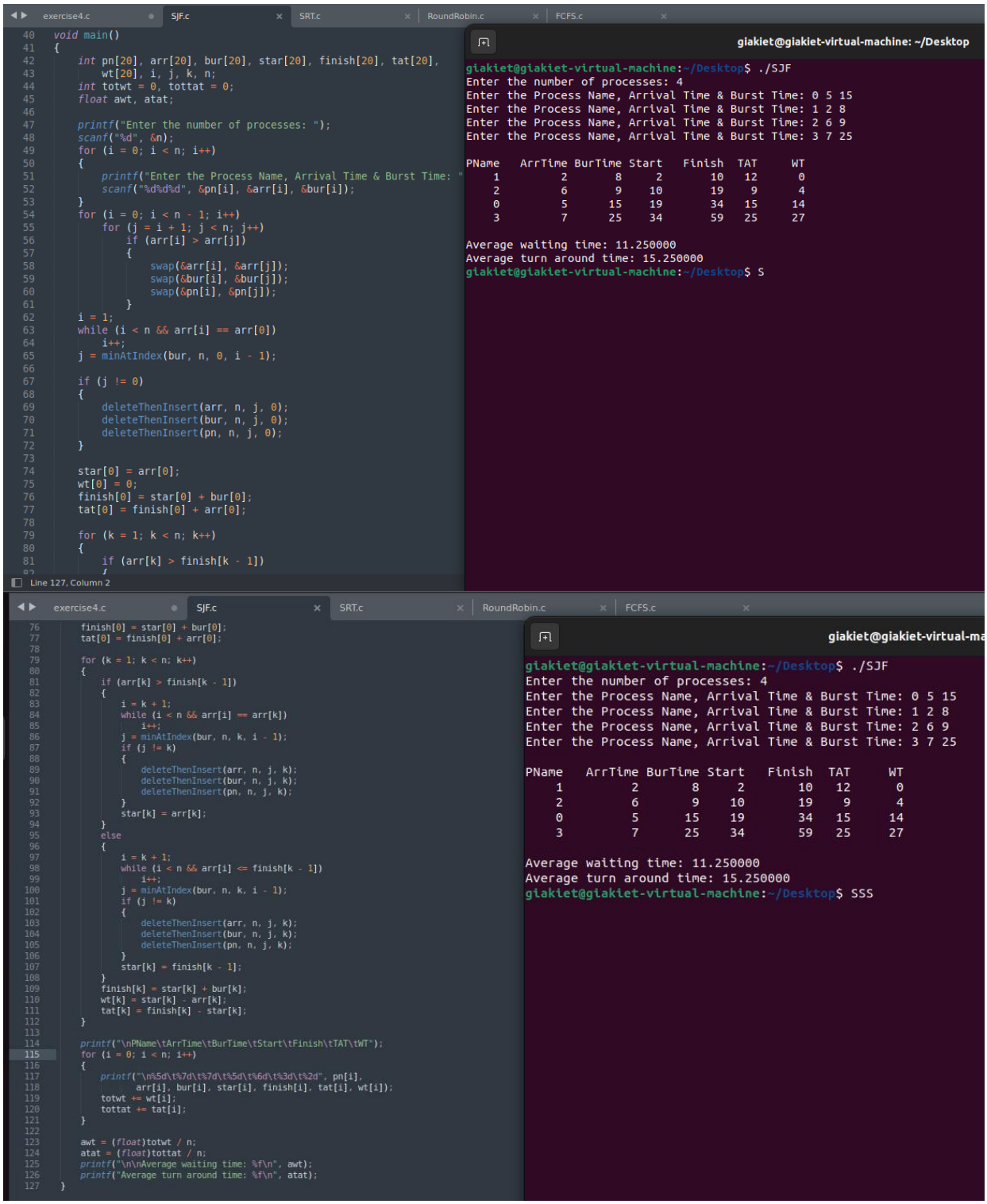
   ❖ Nhập process name, arrival time, burst time

   ❖ In ra Process name, response time, waiting time, turnaround time, average waiting time, average turnaround time.



```
#include <stdio.h>

void swap(int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}

void deleteThenInsert(int a[], int n, int cu, int moi)
{
    int temp = a[cu];
    if (cu < 0 || moi < 0 || cu >= n || moi >= n || moi == cu)
        return;
    else if (moi < cu)
    {
        for (int i = cu; i > moi; i--)
            a[i] = a[i - 1];
        a[moi] = temp;
    }
    else
    {
        for (int i = cu; i < moi; i++)
            a[i] = a[i + 1];
        a[moi] = temp;
    }
}

int minAtIndex(int a[], int n, int start, int end)
{
    if (start > end || start < 0 || end >= n)
        return -1;
    int index = start;
    for (int i = start + 1; i <= end; i++)
        if (a[i] < a[index])
            index = i;
    return index;
}

void main()
{
    int pn[20], arr[20], bur[20], star[20], finish[20], tat[20],
```

```
giakiet@giakiet-virtual-machine:~/Desktop$ ./SJF
Enter the number of processes: 4
Enter the Process Name, Arrival Time & Burst Time: 0 5 15
Enter the Process Name, Arrival Time & Burst Time: 1 2 8
Enter the Process Name, Arrival Time & Burst Time: 2 6 9
Enter the Process Name, Arrival Time & Burst Time: 3 7 25

PName   ArrTime BurTime Start   Finish  TAT     WT
    1         2       8     2       10   12      0
    2         6       9    10       19    9      4
    0         5      15    19       34   15     14
    3         7      25    34       59   25     27

Average waiting time: 11.250000
Average turn around time: 15.250000
giakiet@giakiet-virtual-machine:~/Desktop$
```



2. Viết chương trình mô phỏng giải thuật SRT với các yêu cầu sau:

- ❖ Nhập số lượng process
- ❖ Nhập process name, arrival time, burst time
- ❖ In ra Process name, response time, waiting time, turnaround time, average waiting time, average turnaround time.

exercise4.c | SJF.c | SRT.c | Round

```c
#include <stdio.h>

void main()
{
    int a[10], b[10], x[10];
    int waiting[10], turnaround[10], completion[10];
    int i, j, smallest, count = 0, time, n;
    double avg = 0, tt = 0, end;
    printf("\nEnter the number of Processes: ");
    scanf("%d", &n);
    for (i = 0; i < n; i++)
    {
        printf("Enter arrival, burst time of process %d: ", i + 1);
        scanf("%d", &a[i]);
        scanf("%d", &b[i]);
    }
    for (i = 0; i < n; i++)
    {
        x[i] = b[i];
    }
    b[9] = 9999;
    for (time = 0; count != n; time++)
    {
        smallest = 9;
        for (i = 0; i < n; i++)
        {
            if (a[i] <= time && b[i] < b[smallest] && b[i] > 0)
                smallest = i;
        }
        b[smallest]--;
        if (b[smallest] == 0)
        {
            count++;
            end = time + 1;
            completion[smallest] = end;
            waiting[smallest] = end - a[smallest] - x[smallest];
            turnaround[smallest] = end - a[smallest];
        }
    }
    printf("pid \t burst \t arrival \twaiting \tturnaround \tcompletion");
    for (i = 0; i < n; i++)
    {
        printf("\n %d \t %d \t %d\t\t%d\t\t%d\t\t%d", i + 1,
               x[i], a[i], waiting[i], turnaround[i], completion[i]);
        avg = avg + waiting[i];
        tt = tt + turnaround[i];
    }
    printf("\n\nAverage waiting time: %lf\n", avg / n);
    printf("Average finish time: %lf\n", tt / n);
}
```

giakiet@giakiet-virtual-machine: ~/Desktop

```
giakiet@giakiet-virtual-machine:~/Desktop$ gcc SRT.c -o SRT
SRT.c: In function ‘main’:
SRT.c:43:16: warning: too many arguments for format [-Wformat-extra-args]
   43 |         printf("\n %d \t %d \t %d\t\t%d\t\t%d", i + 1,
      |                ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
giakiet@giakiet-virtual-machine:~/Desktop$ gcc SRT.c -o SRT
SRT.c: In function ‘main’:
SRT.c:43:16: warning: too many arguments for format [-Wformat-extra-args]
   43 |         printf("\n %d \t %d \t %d\t\t%d\t\t%d", i + 1,
      |                ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
giakiet@giakiet-virtual-machine:~/Desktop$ gcc SRT.c -o SRT
giakiet@giakiet-virtual-machine:~/Desktop$ ./SRT

Enter the number of Processes: 6
Enter arrival, burst time of process 1: 0 20
Enter arrival, burst time of process 2: 25 25
Enter arrival, burst time of process 3: 20 25
Enter arrival, burst time of process 4: 35 15
Enter arrival, burst time of process 5: 10 35
Enter arrival, burst time of process 6: 15 50
pid      burst   arrival         waiting         turnaround      completion
 1       20      0               0               20              20
 2       25      25              35              60              85
 3       25      20              0               25              45
 4       15      35              10              25              60
 5       35      10              75              110             120
 6       50      15              105             155             170

Average waiting time: 37.500000
Average finish time: 65.833333
giakiet@giakiet-virtual-machine:~/Desktop$
```

3. Viết chương trình mô phỏng giải thuật RR với các yêu cầu sau (giả sử tất cả các tiến trình đều có arrival time là 0):

- ❖ Nhập số process, Nhập quantum time, Nhập process name, burst time
- ❖ In ra Gantt chart với các thông số: process name, start processor time, stop processor time.
- ❖ In ra average waiting time và average turnaround time.

RoundRobin.c

```c
#include <stdio.h>
int main()
{
    int count, j, n, time, remain, flag = 0, time_quantum;
    int wait_time = 0, turnaround_time = 0;
    int at[10], bt[10], rt[10];
    printf("Enter number of processes:\t ");
    scanf("%d", &n);
    remain = n;
    for (count = 0; count < n; count++)
    {
        printf("Enter Busrt time for Process Number %d: ", count + 1);
        at[count] = 0;
        scanf("%d", &bt[count]);
        rt[count] = bt[count];
    }
    printf("Enter Time quantum:\t");
    scanf("%d", &time_quantum);
    printf("\n\nProcess\t|   Stop Time   |  Start Time\n\n");
    for (time = 0, count = 0; remain != 0;)
    {
        if (rt[count] <= time_quantum && rt[count] > 0)
        {
            int starttime;
            starttime = time;
            time += rt[count];
            rt[count] = 0;
            flag = 1;
            printf("P[%d]\t|\t%d\t|\t%d\n", count + 1, time - at[count], starttime);
        }
        else if (rt[count] > 0)
        {
            rt[count] -= time_quantum;
            time += time_quantum;
            printf("P[%d]\t|\t%d\t|\t%d\n", count + 1, time - at[count], time_quantum);
        }
        if (rt[count] == 0 && flag == 1)
        {
            remain--;
            wait_time += time - at[count] - bt[count];
            turnaround_time += time - at[count];
            flag = 0;
        }
        if (count == n - 1)
            count = 0;
        else if (at[count + 1] <= time)
            count++;
        else
            count = 0;
    }
    printf("\nAverage waiting time: %f\n", wait_time * 1.0 / n);
    printf("Average turn around time: %f\n", turnaround_time * 1.0 / n);
    printf("\n");
    return 0;
}
```

giakiet@giakiet-virtual-machine: ~/Desktop

```
giakiet@giakiet-virtual-machine:~/Desktop$ gcc RoundRobin.c -o RR
giakiet@giakiet-virtual-machine:~/Desktop$ ./RR
Enter number of processes:      6
Enter Busrt time for Process Number 1: 20
Enter Busrt time for Process Number 2: 25
Enter Busrt time for Process Number 3: 25
Enter Busrt time for Process Number 4: 15
Enter Busrt time for Process Number 5: 35
Enter Busrt time for Process Number 6: 50
Enter Time quantum:     10


Process |   Stop Time   |  Start Time

P[1]    |       10      |       10
P[2]    |       20      |       10
P[3]    |       30      |       10
P[4]    |       40      |       10
P[5]    |       50      |       10
P[6]    |       60      |       10
P[1]    |       70      |       60
P[2]    |       80      |       10
P[3]    |       90      |       10
P[4]    |       95      |       90
P[5]    |       105     |       10
P[6]    |       115     |       10
P[2]    |       120     |       115
P[3]    |       125     |       120
P[5]    |       135     |       10
P[6]    |       145     |       10
P[5]    |       150     |       145
P[6]    |       160     |       10
P[6]    |       170     |       160

Average waiting time: 93.333333
Average turn around time: 121.666667

giakiet@giakiet-virtual-machine:~/Desktop$
```