Name : Mandhani Vaibhav Ramesh Sub : OS

Roll N0 : 266 Practical : 02

**FCFS Scheduling**

#include <stdio.h>

*// key = 0: sort by arrival time*

*// key = 1: sort by process number*

void sort(int n, int key, int process\_no[], int at[], int bt[], int ct[], int ta[], int wt[]) {

    int temp;

    for(int i = 0; i < n; i++) {

        for(int j = i + 1; j < n; j++) {

            int condition = 0;

            if (key == 0) {

                condition = (at[i] > at[j]); *// sort by arrival time*

            } else if (key == 1) {

                condition = (process\_no[i] > process\_no[j]); *// sort by process number*

            }

            if (condition) {

                #define SWAP(x, y) { temp = x; x = y; y = temp; }

                SWAP(process\_no[i], process\_no[j]);

                SWAP(at[i], at[j]);

                SWAP(bt[i], bt[j]);

                SWAP(ct[i], ct[j]);

                SWAP(ta[i], ta[j]);

                SWAP(wt[i], wt[j]);

            }

        }

    }

}

void calculate(int n, int at[], int bt[], int ta[], int wt[], int ct[]) {

    int time = 0;

    for (int i = 0; i < n; i++) {

        if (time < at[i]) {

            time = at[i];

        }

        ct[i] = time + bt[i];

        ta[i] = ct[i] - at[i];

        wt[i] = ta[i] - bt[i];

        time = ct[i];

    }

}

void main() {

    int n;

    printf("FCFS scheduling!\n");

    printf("Enter no of processes: ");

    scanf("%d", &n);

    int process\_no[n], at[n], bt[n], ta[n], wt[n], ct[n];

    for (int i = 0; i < n; i++) {

        process\_no[i] = i + 1;

        printf("\nEnter arrival time of process %d: ", i + 1);

        scanf("%d", &at[i]);

        printf("Enter burst time of process %d: ", i + 1);

        scanf("%d", &bt[i]);

        ct[i] = ta[i] = wt[i] = 0;

    }

    sort(n, 0, process\_no, at, bt, ct, ta, wt);

    printf("\nSorted Processes according to arrival time:");

    printf("\nProcess No\tAT\tBT");

    for (int i = 0; i < n; i++) {

        printf("\n%d\t\t%d\t%d", process\_no[i], at[i], bt[i]);

    }

    calculate(n, at, bt, ta, wt, ct);

    sort(n, 1, process\_no, at, bt, ct, ta, wt);

    float avg\_ta = 0, avg\_wt = 0;

    for (int i = 0; i < n; i++) {

        avg\_ta += ta[i];

        avg\_wt += wt[i];

    }

    printf("\n FCFS Scheduling Table....:");

    printf("\nProcess No\tAT\tBT\tCT\tTA\tWT");

    for (int i = 0; i < n; i++) {

        printf("\n%d\t\t%d\t%d\t%d\t%d\t%d", process\_no[i], at[i], bt[i], ct[i], ta[i], wt[i]);

    }

    printf("\nAverage TA: %.2f",avg\_ta / n);

    printf("\nAverage WT: %.2f\n",avg\_wt / n);

}

**Ouput :**

1. **Arrival Time in order**

D:\MGM\4TH\_SEM\OS>gcc fcfs.c

D:\MGM\4TH\_SEM\OS>a.exe

FCFS scheduling!

Enter no of processes: 4

Enter arrival time of process 1: 0

Enter burst time of process 1: 4

Enter arrival time of process 2: 1

Enter burst time of process 2: 3

Enter arrival time of process 3: 2

Enter burst time of process 3: 1

Enter arrival time of process 4: 3

Enter burst time of process 4: 2

Sorted Processes according to arrival time:

Process No AT BT

1 0 4

2 1 3

3 2 1

4 3 2

FCFS Scheduling Table....:

Process No AT BT CT TA WT

1 0 4 4 4 0

2 1 3 7 6 3

3 2 1 8 6 5

4 3 2 10 7 5

Average TA: 5.75

Average WT: 3.25

1. **Arrival time in disorder**

D:\MGM\4TH\_SEM\OS>a.exe

FCFS scheduling!

Enter no of processes:

4

Enter arrival time of process 1: 2

Enter burst time of process 1: 3

Enter arrival time of process 2: 0

Enter burst time of process 2: 5

Enter arrival time of process 3: 1

Enter burst time of process 3: 2

Enter arrival time of process 4: 3

Enter burst time of process 4: 4

Sorted Processes according to arrival time:

Process No AT BT

2 0 5

3 1 2

1 2 3

4 3 4

FCFS Scheduling Table....:

Process No AT BT CT TA WT

1 2 3 10 8 5

2 0 5 5 5 0

3 1 2 7 6 4

4 3 4 14 11 7

Average TA: 7.50

Average WT: 4.00

1. **With CPU idle time**

D:\MGM\4TH\_SEM\OS>a.exe

FCFS scheduling!

Enter no of processes: 4

Enter arrival time of process 1: 0

Enter burst time of process 1: 3

Enter arrival time of process 2: 6

Enter burst time of process 2: 2

Enter arrival time of process 3: 5

Enter burst time of process 3: 4

Enter arrival time of process 4: 9

Enter burst time of process 4: 5

Sorted Processes according to arrival time:

Process No AT BT

1 0 3

3 5 4

2 6 2

4 9 5

FCFS Scheduling Table....:

Process No AT BT CT TA WT

1 0 3 3 3 0

2 6 2 11 5 3

3 5 4 9 4 0

4 9 5 16 7 2

Average TA: 4.75

Average WT: 1.25

D:\MGM\4TH\_SEM\OS>