**LAB – 7**

**AIM –** **Disk Scheduling algorithm (FCFS, SSTF)**

SHRUTI MISHRA

21BCP110

**FCFS:**

**CODE:**

#include <stdio.h>

int main() {

    int head, n, s = 0;

    int queue[100], ans[100];

    printf("Enter the value of head:");

    scanf("%d", &head);

    queue[0] = head;

    printf("Enter the number of processes:");

    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {

        int x;

        printf("Enter the value of process:");

        scanf("%d", &x);

        queue[i] = x;

    }

    printf("[");

    for (int i = 0; i <= n; i++) {

        printf("%d, ", queue[i]);

    }

    printf("]\n");

    for (int i = 0; i < n; i++) {

        int j;

        if (queue[i+1] > queue[i]) {

            j = queue[i+1] - queue[i];

        } else {

            i++;

            j = queue[i+1] - queue[i];

        }

        s += j;

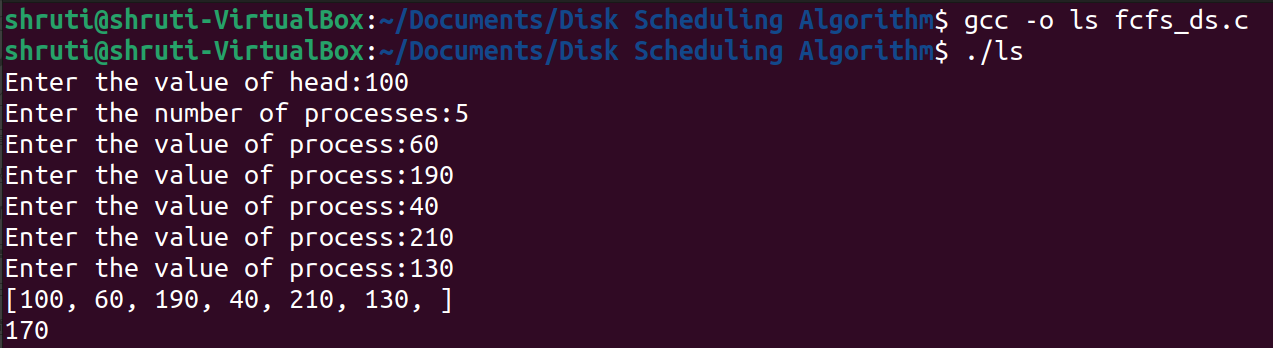
    }

    printf("%d\n", s);

    return 0;

}

**OUTPUT**:



**SSTF:**

**CODE**

#include <stdio.h>

int main() {

    int head, queue[100], temp[100];

    int s = 0, n, i, x, ans, val, val1;

    printf("Enter the value of head:");

    scanf("%d", &head);

    printf("Enter the number of processes:");

    scanf("%d", &n);

    for (i = 0; i < n; i++) {

        printf("Enter the value of process:");

        scanf("%d", &x);

        queue[i] = x;

    }

    printf("Queue: ");

    for (i = 0; i < n; i++) {

        printf("%d ", queue[i]);

    }

    printf("\n");

    int available[n];

    for (i = 0; i < n; i++) {

        available[i] = queue[i];

    }

    for (i = 0; i < n; i++) {

        int len = n - i;

        for (int j = 0; j < len; j++) {

            if (head > available[j]) {

                ans = head - available[j];

            } else {

                ans = available[j] - head;

            }

            temp[j] = ans;

        }

        printf("Available: ");

        for (int k = 0; k < len; k++) {

            printf("%d ", available[k]);

        }

        printf("\n");

        printf("Temp: ");

        for (int k = 0; k < len; k++) {

            printf("%d ", temp[k]);

        }

        printf("\n");

        val1 = 0;

        for (int j = 0; j < len; j++) {

            if (temp[j] < temp[val1]) {

                val1 = j;

            }

        }

        val = val1 + 1;

        s = s + temp[val1];

        head = available[val1];

        for (int j = val1; j < len - 1; j++) {

            available[j] = available[j+1];

        }

    }

    printf("Total seek time: %d\n", s);

    return 0;

}

**OUTPUT**:

