EXPERIMENT-10

Aim of the experiment: disk scheduling algorithm

Q: FCFS disk scheduling algorithm using C.

Input:

```c
#include <stdio.h>
#include <stdlib.h>

int main() {
    int requestQueue[100];
    int initialHeadPosition, totalHeadMovement = 0, numRequests;
    int currentHeadPosition;

    printf("Enter the number of requests: ");
    scanf("%d", &numRequests);

    printf("Enter the initial head position: ");
    scanf("%d", &initialHeadPosition);
    currentHeadPosition = initialHeadPosition;

    printf("Enter the request queue: ");
    for (int i = 0; i < numRequests; i++) {
        scanf("%d", &requestQueue[i]);
    }

    printf("\nFCFS Disk Scheduling:\n");
    printf("Sequence of Head Movement: %d", initialHeadPosition);

    for (int i = 0; i < numRequests; i++) {
        int nextRequest = requestQueue[i];
        int movement = abs(nextRequest - currentHeadPosition);
        totalHeadMovement += movement;
        currentHeadPosition = nextRequest;
        printf(" -> %d", currentHeadPosition);
    }

    printf("\n\nInitial Head Position: %d\n", initialHeadPosition);
    printf("Total Head Movement: %d\n", totalHeadMovement);

    return 0;
}
```

Output:

```
piyush@piyush-VirtualBox:~$ gcc first.c -o first
piyush@piyush-VirtualBox:~$ ./first
Enter the number of requests: 5
Enter the initial head position: 20
Enter the request queue: 8 170 43 120 24

FCFS Disk Scheduling:
Sequence of Head Movement: 20 -> 8 -> 170 -> 43 -> 120 -> 24

Initial Head Position: 20
Total Head Movement: 474
```

Q: Scan disk scheduling using C.

Input:

```c
#include <stdio.h>
#include <stdlib.h>

void sort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

int main() {
    int requestQueue[100];
    int initialHeadPosition, totalHeadMovement = 0, numRequests;
    int diskSize = 199;
    int direction = 1;

    printf("Enter the number of requests: ");
    scanf("%d", &numRequests);

    printf("Enter the initial head position: ");
    scanf("%d", &initialHeadPosition);

    printf("Enter the request queue: ");
    for (int i = 0; i < numRequests; i++) {
        scanf("%d", &requestQueue[i]);
    }

    printf("Enter the maximum cylinder number (e.g., 199): ");
    scanf("%d", &diskSize);

    int sequence[102];
    for(int i = 0; i < numRequests; i++) {
        sequence[i] = requestQueue[i];
    }
    sequence[numRequests] = initialHeadPosition;
    sequence[numRequests + 1] = diskSize;
    int sequenceSize = numRequests + 2;

    sort(sequence, sequenceSize);

    int start_index = -1;
    for (int i = 0; i < sequenceSize; i++) {
        if (sequence[i] == initialHeadPosition) {
            start_index = i;
            break;
        }

    }
}

printf("\nSCAN Disk Scheduling (Head moves right first):\n");
printf("Sequence of Head Movement: %d", initialHeadPosition);
int currentHeadPosition = initialHeadPosition;

for (int i = start_index + 1; i < sequenceSize; i++) {
    int nextRequest = sequence[i];
    totalHeadMovement += abs(nextRequest - currentHeadPosition);
    currentHeadPosition = nextRequest;
    printf(" -> %d", currentHeadPosition);
}

for (int i = start_index - 1; i >= 0; i--) {
    int nextRequest = sequence[i];
    totalHeadMovement += abs(nextRequest - currentHeadPosition);
    currentHeadPosition = nextRequest;
    printf(" -> %d", currentHeadPosition);
}

printf("\n\nInitial Head Position: %d\n", initialHeadPosition);
printf("Total Head Movement: %d\n", totalHeadMovement);

return 0;
}
```

Output:

```
piyush@piyush-VirtualBox:~$ gcc second.c -o second
piyush@piyush-VirtualBox:~$ ./second
Enter the number of requests: 7
Enter the initial head position: 20
Enter the request queue: 8 170 43 140 24 90 125
Enter the maximum cylinder number (e.g., 199): 199

SCAN Disk Scheduling (Head moves right first):
Sequence of Head Movement: 20 -> 24 -> 43 -> 90 -> 125 -> 140 -> 170 -> 199 -> 8

Initial Head Position: 20
Total Head Movement: 370
```

Q.CSCAN algorithm using c.

Input:

```c
#include <stdio.h>
#include <stdlib.h>

void sort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

int main() {
    int requestQueue[100];
    int initialHeadPosition, totalHeadMovement = 0, numRequests;
    int diskSize = 199;
    int direction = 1;

    printf("Enter the number of requests: ");
    scanf("%d", &numRequests);

    printf("Enter the initial head position: ");
    scanf("%d", &initialHeadPosition);

    printf("Enter the request queue: ");
    for (int i = 0; i < numRequests; i++) {
        scanf("%d", &requestQueue[i]);
    }

    printf("Enter the maximum cylinder number (e.g., 199): ");
    scanf("%d", &diskSize);

    int sequence[102];
    for(int i = 0; i < numRequests; i++) {
        sequence[i] = requestQueue[i];
    }
    sequence[numRequests] = initialHeadPosition;
    sequence[numRequests + 1] = diskSize;
    sequence[numRequests + 2] = 0;
    int sequenceSize = numRequests + 3;

    sort(sequence, sequenceSize);

    int start_index = -1;
    for (int i = 0; i < sequenceSize; i++) {
        if (sequence[i] == initialHeadPosition) {
            start_index = i;
            break;
```

```
            break;
        }
    }

    printf("\nC-SCAN Disk Scheduling (Head moves right):\n");
    printf("Sequence of Head Movement: %d", initialHeadPosition);
    int currentHeadPosition = initialHeadPosition;

    for (int i = start_index + 1; i < sequenceSize; i++) {
        int nextRequest = sequence[i];
        if (nextRequest == 0) continue;
        totalHeadMovement += abs(nextRequest - currentHeadPosition);
        currentHeadPosition = nextRequest;
        printf(" -> %d", currentHeadPosition);
    }

    if (currentHeadPosition != diskSize) {
        totalHeadMovement += abs(diskSize - currentHeadPosition);
        currentHeadPosition = diskSize;
        printf(" -> %d", currentHeadPosition);
    }

    totalHeadMovement += abs(currentHeadPosition - 0);
    currentHeadPosition = 0;
    printf(" -> %d (Wrap-Around)", currentHeadPosition);

    for (int i = 0; i < start_index; i++) {
        int nextRequest = sequence[i];
        totalHeadMovement += abs(nextRequest - currentHeadPosition);
        currentHeadPosition = nextRequest;
        printf(" -> %d", currentHeadPosition);
    }

    printf("\n\nInitial Head Position: %d\n", initialHeadPosition);
    printf("Total Head Movement: %d\n", totalHeadMovement);

    return 0;
}
```

Output:

```
piyush@piyush-VirtualBox:~$ gcc third.c -o third
piyush@piyush-VirtualBox:~$ ./third
Enter the number of requests: 9
Enter the initial head position: 20
Enter the request queue: 8 170 147 129 110 65 47 89
90
Enter the maximum cylinder number (e.g., 199): 199

C-SCAN Disk Scheduling (Head moves right):
Sequence of Head Movement: 20 -> 47 -> 65 -> 89 -> 90 -> 110 -> 129 -> 147 -> 17
0 -> 199 -> 0 (Wrap-Around) -> 0 -> 8

Initial Head Position: 20
Total Head Movement: 386
```


SUBMITTED BY :

NAME : PIYUSH SUTAR

REGD NO : 2301020880

GROUP : 6

SEM : 5TH

BRANCH : CSE(AI & ML)