Fr. Conceicao Rodrigues College of Engineering Department of Computer Engineering			
Student's Roll No	**9558**	**Students Name**	**Brita**
Date of Performance		**SE Computer – Div**	**A**

Aim: Study Disk Management

Lab Outcome:

CSL403.6: Implement various Disk Management techniques and evaluate their performance.

Problem Statements:
Implement Disk Management Algorithms

(a)FCFS (b)SSTF (c)SCAN

Given the current head position and future disk block references wrt tracks or cylinders. Calculate the seek length based on above algorithms. Show the sequence in which the disk blocks will be accessed and no of tracks traversed in each algorithm.

FCFS

CODE:

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int main()
{
   int n, head_pos, seek_length = 0, disk_size;
   printf("Enter the size of the disk: ");
   scanf("%d", &disk_size);
   printf("Enter number of requests: ");
   scanf("%d", &n);
   int requests[n];
   printf("Enter the requests: ");
   for (int i = 0; i < n; i++)
   {
    scanf("%d", &requests[i]);
    if (requests[i] < 0 || requests[i] > disk_size)
   {
    printf("Error: Request is outside the range of the disk\n", i+1);
    return 0;
   }
```

```
  }
   printf("Enter the head position: ");
   scanf("%d", &head_pos);
   if (head_pos < 0 || head_pos > disk_size)
  {
   printf("Error: Head position is outside the range of the disk\n");
   return 0;
  }
   printf("Head position: %d\n", head_pos);
   printf("Sequence: %d -> ", head_pos);
   for (int i = 0; i < n; i++)
   {
   printf("%d -> ", requests[i]);
   seek_length += abs(requests[i] - head_pos);
    head_pos = requests[i];
    }
   printf("\n");
   printf("Total seek length: %d\n", seek_length);
   return 0;
   }
```

OUTPUT:

Enter the size of the disk: 300
Enter number of requests: 7
Enter the requests: 82 170 43 140 24 16 190
Enter the head position: 50
Head position: 50
Sequence: 50 -> 82 -> 170 -> 43 -> 140 -> 24 -> 16 -> 190 ->
Total seek length: 642

Enter the size of the disk: 300
Enter number of requests: 3
Enter the requests: 100 500 43
Error: Request is outside the range of the disk

Enter the size of the disk: 300
Enter number of requests: 7
Enter the requests: 10 90 53 49 43 16 30
Enter the head position: 500
Error: Head position is outside the range of the disk

SSTF

CODE :

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

int main()
{
   int n, head_pos, seek_length = 0, disk_size;
   printf("Enter the size of the disk: ");
   scanf("%d", &disk_size);
   printf("Enter number of requests: ");
   scanf("%d", &n);
   int requests[n], done[n];
   printf("Enter the requests:");
   for (int i = 0; i < n; i++)
   {
    done[i] = 0;
    scanf("%d", &requests[i]);
    if (requests[i] < 0 || requests[i] > disk_size)
   {
    printf("Error: Request is outside the range of the disk\n", i+1);
    return 0;
      }
   }
   printf("Enter the head position: ");
   scanf("%d", &head_pos);
   if (head_pos < 0 || head_pos > disk_size) {
    printf("Error: Head position is outside the range of the disk\n");
    return 0;
   }
   printf("Head position: %d\n", head_pos);
   printf("Sequence: %d -> ", head_pos);
   for (int count = 0; count < n; count++)
  {
     int min_dist = disk_size + 1, min_idx = 0;
     for (int j = 0; j < n; j++)
    {
```

```
      if (done[j] == 0)
    {
       int dist = abs(requests[j] - head_pos);
       if (dist < min_dist)
      {
        min_dist = dist;
        min_idx = j;
            }
          }
       }
       done[min_idx] = 1;
       seek_length += min_dist;
       head_pos = requests[min_idx];
       printf("%d -> ", requests[min_idx]);
    }
    printf("\n");
    printf("Total seek length: %d\n", seek_length);
    return 0;
}
```

OUTPUT:

```
Enter the size of the disk: 300
Enter number of requests: 7
Enter the requests:82 170 43 140 24 16 190
Enter the head position: 50
Head position: 50
Sequence: 50 -> 43 -> 24 -> 16 -> 82 -> 140 -> 170 -> 190 ->
Total seek length: 208

Enter the size of the disk: 300
Enter number of requests: 3
Enter the requests:100 99 360
Error: Request is outside the range of the disk

Enter the size of the disk: 300
Enter number of requests: 3
Enter the requests:100 200 300
Enter the head position: -90
Error: Head position is outside the range of the disk
```

SCAN

CODE:

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

int main()
{
   int n, head_pos, seek_length = 0, disk_size;
   printf("Enter the size of the disk: ");
   scanf("%d", &disk_size);
   printf("Enter number of requests: ");
   scanf("%d", &n);
   int requests[n];
   printf("Enter the requests: ");
   for (int i = 0; i < n; i++)
   {
   scanf("%d", &requests[i]);
   if (requests[i] < 0 || requests[i] > disk_size)
   {
    printf("Error: Request is outside the range of the disk\n", i+1);
    return 0;
      }
   }

   printf("Enter the head position: ");
   scanf("%d", &head_pos);
   if (head_pos < 0 || head_pos > disk_size)
  {
   printf("Error: Head position is outside the range of the disk\n");
   return 0;
   }
   printf("Head position: %d\n", head_pos);
   printf("Sequence: %d -> ", head_pos);
```

```
// Sorting  the requests in ascending order
   for (int i = 0; i < n; i++)
   {
    for (int j = i+1; j < n; j++)
   {
    if (requests[i] > requests[j])
   {
       int temp = requests[i];
       requests[i] = requests[j];
       requests[j] = temp;
          }
      }
   }

   // Finding the index of the first request that is larger than the head position
   int start_index = 0;
   for (int i = 0; i < n; i++)
   {
   if (requests[i] >= head_pos)
   {
   start_index = i;
   break;
      }
   }

   // Go in the direction of increasing request values until the last request
   for (int i = start_index; i < n; i++)
   {
      printf("%d -> ", requests[i]);
      seek_length += abs(requests[i] - head_pos);
      head_pos = requests[i];
   }

   // Go in the direction of decreasing request values until the first request
   for (int i = start_index-1; i >= 0; i--)
   {
      printf("%d -> ", requests[i]);
      seek_length += abs(requests[i] - head_pos);
      head_pos = requests[i];
```

```
    }
    printf("\n");
    printf("Total seek length: %d\n", seek_length);
    return 0;
}
```

OUTPUT:

```
Enter the size of the disk: 300
Enter number of requests: 7
Enter the requests: 82 170 43 140 24 16 190
Enter the head position: 50
Head position: 50
Sequence: 50 -> 82 -> 140 -> 170 -> 190 -> 43 -> 24 -> 16 ->
Total seek length: 314

Enter the size of the disk: 300
Enter number of requests: 3
Enter the requests: 120 -12 70
Error: Request is outside the range of the disk

Enter the size of the disk: 700
Enter number of requests: 7
Enter the requests: 100 99 120 300 500 412 34
Enter the head position: 750
Error: Head position is outside the range of the disk
```

On time Submission(2)	Knowledge of Topic(4)	Implementation and Demonstraion(4)	Total (10)

Signature of Faculty		**Date of Submission**	