Operating Systems
CT-353

Name: Sabrina Shahzad
Roll No. : DT-026

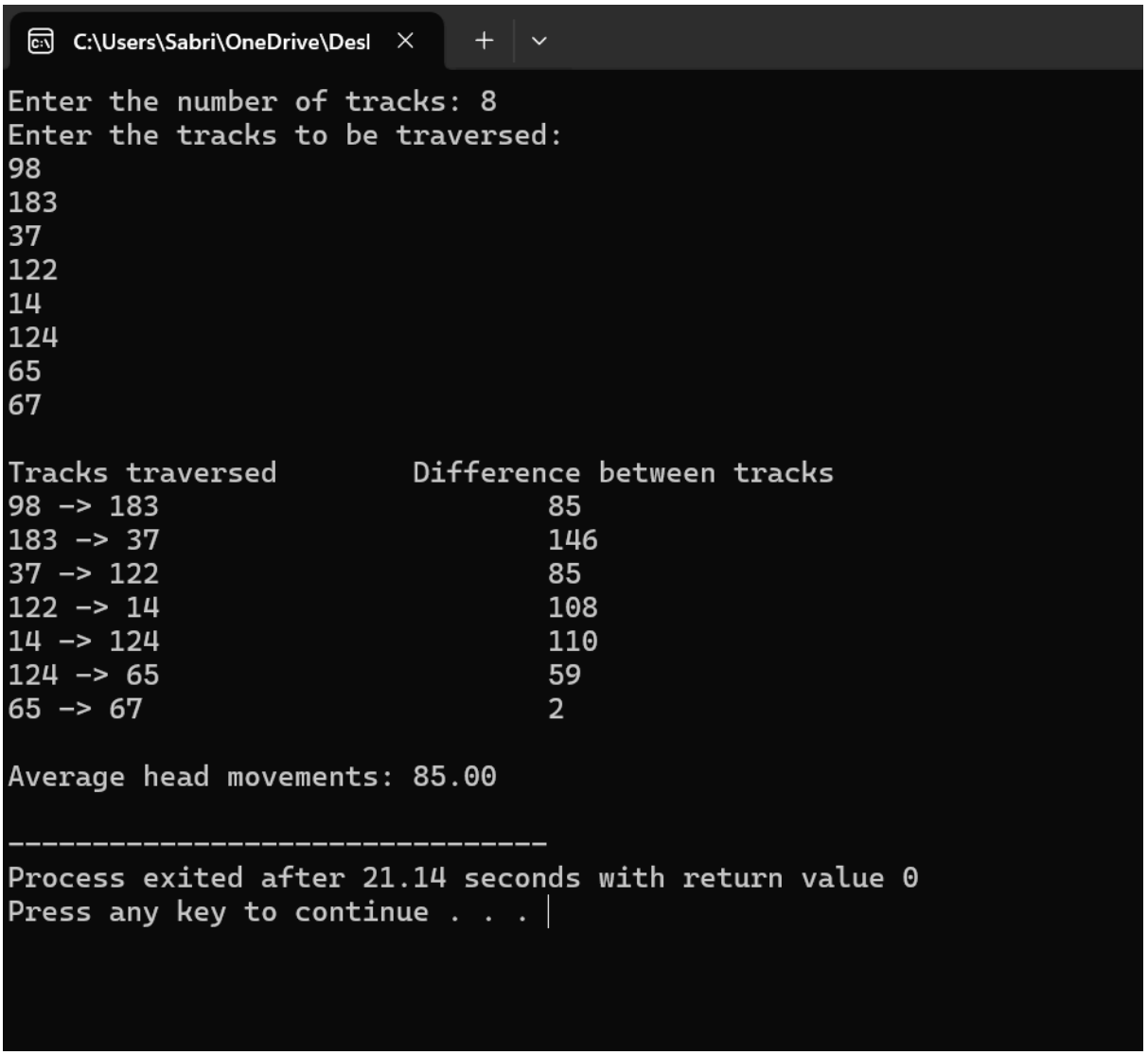
Lab 13:
DISK SCHEDULING ALGORITHMS

(A) FCFS:

```
#include <stdio.h>
#include <stdlib.h>  // for abs()

int main() {
   int t[20], n, i;
   int tohm[20], tot = 0;
   float avhm;

   printf("Enter the number of tracks: ");
   scanf("%d", &n);

   printf("Enter the tracks to be traversed:\n");
   for (i = 0; i < n; i++) {
      scanf("%d", &t[i]);
   }

   for (i = 0; i < n - 1; i++) {
      tohm[i] = abs(t[i + 1] - t[i]);
      tot += tohm[i];
   }

   avhm = (float) tot / (n - 1);

   printf("\nTracks traversed\tDifference between tracks\n");
   for (i = 0; i < n - 1; i++) {
      printf("%d -> %d\t\t\t%d\n", t[i], t[i + 1], tohm[i]);
   }

   printf("\nAverage head movements: %.2f\n", avhm);

   return 0;
```

}

Output:

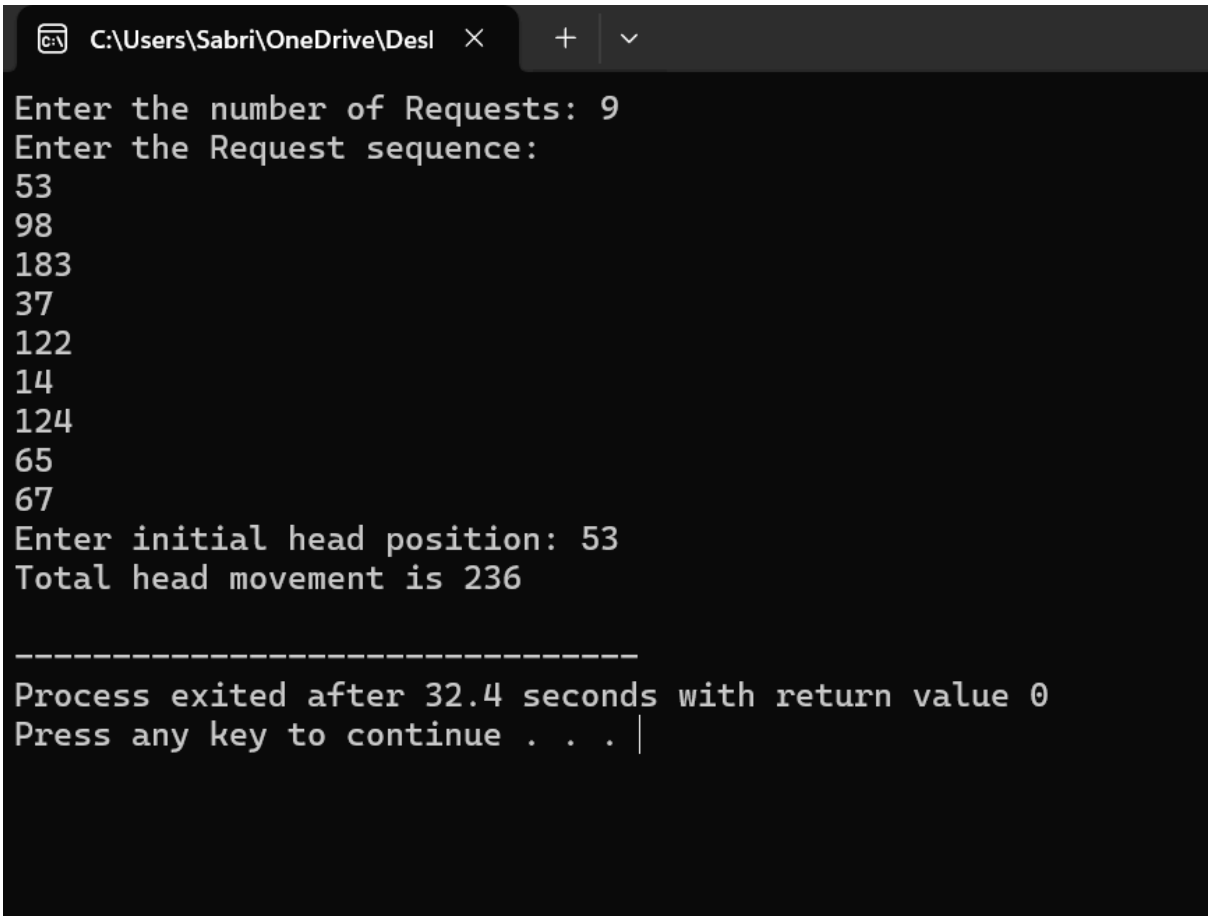

C:\Users\Sabri\OneDrive\Des
Enter the number of tracks: 8
Enter the tracks to be traversed:
98
183
37
122
14
124
65
67
Tracks traversed Difference between tracks
98 -> 183 85
183 -> 37 146
37 -> 122 85
122 -> 14 108
14 -> 124 110
124 -> 65 59
65 -> 67 2
Average head movements: 85.00
Process exited after 21.14 seconds with return value 0
Press any key to continue . . .

(B) SSTF

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

int main() {
   int RQ[100], i, n, TotalHeadMovement = 0, initial, count = 0;
   printf("Enter the number of Requests: ");
   scanf("%d", &n);

   printf("Enter the Request sequence:\n");
   for (i = 0; i < n; i++)
      scanf("%d", &RQ[i]);

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

   // SSTF Disk Scheduling Logic
   while (count != n) {
      int min = 1000, d, index = -1;

      for (i = 0; i < n; i++) {
         d = abs(RQ[i] - initial);
         if (RQ[i] != -1 && min > d) { // check if not already serviced
            min = d;
            index = i;
         }
      }

      TotalHeadMovement += min;
```

```
            initial = RQ[index];
            RQ[index] = -1; // mark as serviced
            count++;
        }

        printf("Total head movement is %d\n", TotalHeadMovement);
        return 0;
    }
```

Output:



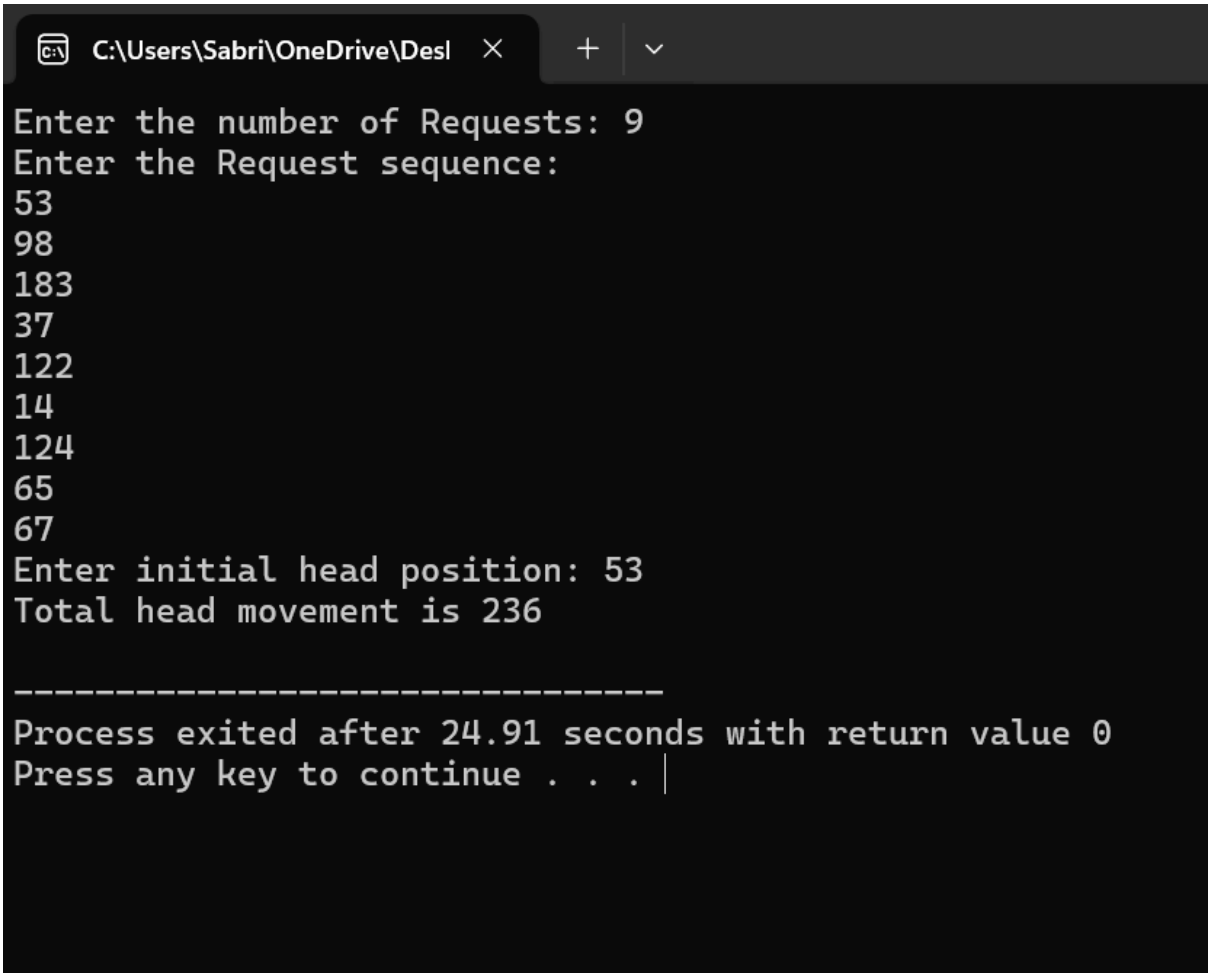
(C) SCAN

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

int main() {
   int t[20], n, head, i, j, temp, pos, total = 0;
   int atr[20], p = 0, max = 199; // assuming disk size is 0 to 199
   float avg;

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

   printf("Enter the track requests:\n");
   for (i = 0; i < n; i++) {
      scanf("%d", &t[i]);
   }

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

   // Add head to track list
   t[n] = head;
   n++;

   // Sort the track requests including the head
   for (i = 0; i < n - 1; i++) {
      for (j = 0; j < n - i - 1; j++) {
         if (t[j] > t[j + 1]) {
            temp = t[j];
```

```
            t[j] = t[j + 1];
            t[j + 1] = temp;
        }
    }
}

// Find the position of head after sorting
for (i = 0; i < n; i++) {
    if (t[i] == head) {
        pos = i;
        break;
    }
}

// Traverse right from head to end
for (i = pos; i < n; i++) {
    atr[p++] = t[i];
}

// Simulate jump to beginning
atr[p++] = max;
atr[p++] = 0;

// Then traverse from start to before head
for (i = 0; i < pos; i++) {
    atr[p++] = t[i];
}

// Calculate total head movements
```

```
    for (i = 0; i < p - 1; i++) {
        total += abs(atr[i + 1] - atr[i]);
    }

    avg = (float)total / (n - 1); // average over actual requests

    printf("\nC-SCAN Order of Service:\n");
    for (i = 0; i < p; i++) {
        printf("%d ", atr[i]);
    }

    printf("\n\nTotal head movement: %d", total);
    printf("\nAverage head movement: %.2f\n", avg);

    return 0;
}
```

Output:


C:\Users\Sabri\OneDrive\Desl
Enter the number of Requests: 9
Enter the Request sequence:
53
98
183
37
122
14
124
65
67
Enter initial head position: 53
Total head movement is 236

Process exited after 24.91 seconds with return value 0
Press any key to continue . . .