

NAME:-PRIYANK JHAVERI

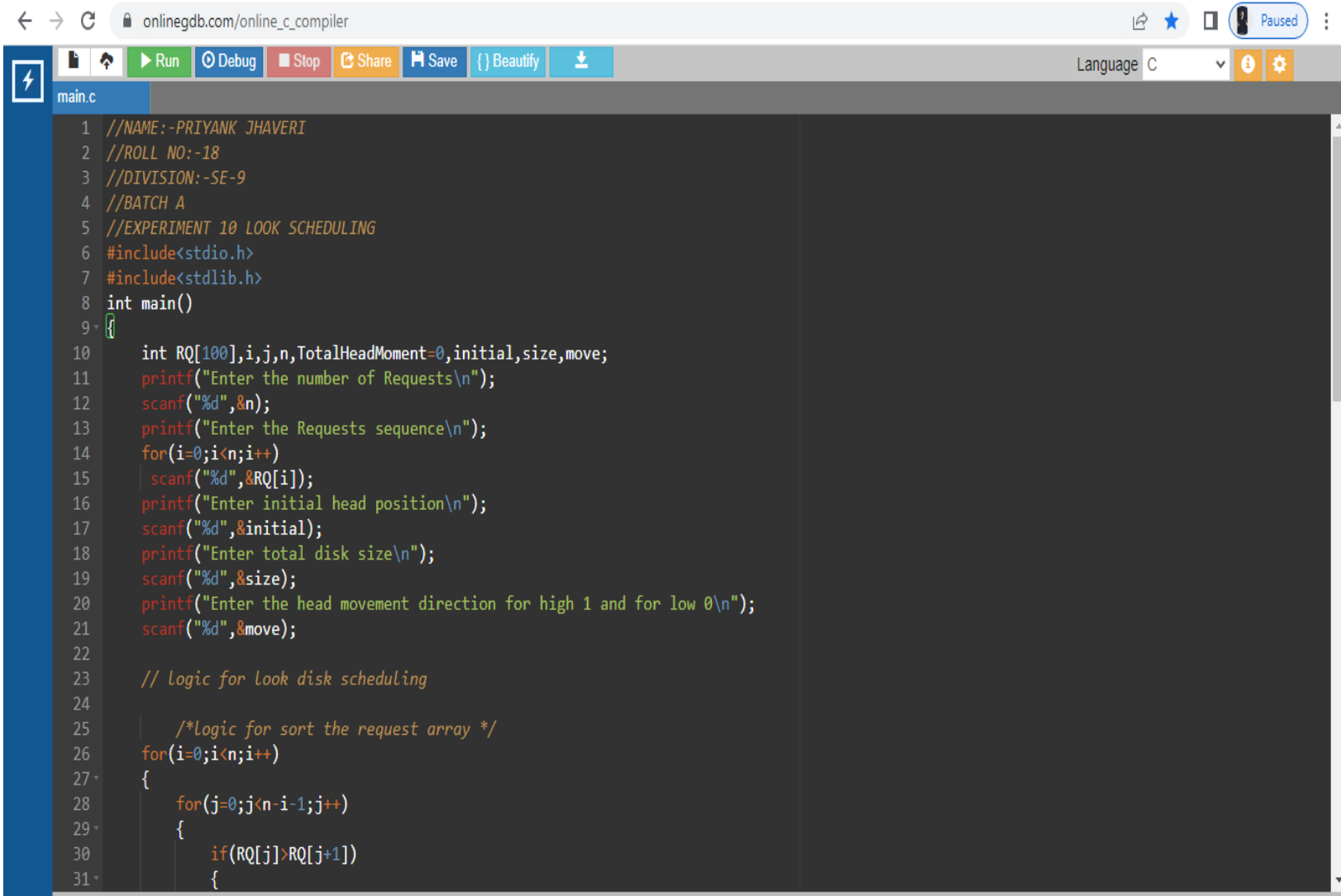
ROLL NO:-18

DIVISION:-SE-9

BATCH A

EXPERIMENT 10

LOOK SCHEDULING



The screenshot shows a web browser window with the URL `onlinegdb.com/online_c_compiler`. The browser's address bar and tabs are visible. Below the browser window is a toolbar with buttons for Run, Debug, Stop, Share, Save, and Beautify. The main area displays a C program for Look Scheduling. The program includes headers for `stdio.h` and `stdlib.h`, and defines a `main` function. It prompts the user to enter the number of requests, the requests sequence, the initial head position, the total disk size, and the head movement direction. The program then implements the Look Scheduling algorithm by sorting the request array and calculating the total head movement.

```
1 //NAME:-PRIYANK JHAVERI
2 //ROLL NO:-18
3 //DIVISION:-SE-9
4 //BATCH A
5 //EXPERIMENT 10 LOOK SCHEDULING
6 #include<stdio.h>
7 #include<stdlib.h>
8 int main()
9 {
10     int RQ[100],i,j,n,TotalHeadMoment=0,initial,size,move;
11     printf("Enter the number of Requests\n");
12     scanf("%d",&n);
13     printf("Enter the Requests sequence\n");
14     for(i=0;i<n;i++)
15     {
16         scanf("%d",&RQ[i]);
17     }
18     printf("Enter initial head position\n");
19     scanf("%d",&initial);
20     printf("Enter total disk size\n");
21     scanf("%d",&size);
22     printf("Enter the head movement direction for high 1 and for low 0\n");
23     scanf("%d",&move);
24
25     // Logic for look disk scheduling
26
27     /*Logic for sort the request array */
28     for(i=0;i<n;i++)
29     {
30         for(j=0;j<n-i-1;j++)
31         {
32             if(RQ[j]>RQ[j+1])
33             {
34                 // Swap RQ[j] and RQ[j+1]
35                 int temp=RQ[j];
36                 RQ[j]=RQ[j+1];
37                 RQ[j+1]=temp;
38             }
39         }
40     }
41
42     // Calculate total head movement
43     if(move==1)
44     {
45         // High to Low
46         for(i=0;i<n;i++)
47         {
48             TotalHeadMoment+=abs(RQ[i]-RQ[i+1]);
49         }
50     }
51     else
52     {
53         // Low to High
54         for(i=0;i<n;i++)
55         {
56             TotalHeadMoment+=abs(RQ[i]-RQ[i+1]);
57         }
58     }
59
60     printf("Total Head Movement = %d",TotalHeadMoment);
61 }
```

NAME:-PRIYANK JHAVERI

ROLL NO:-18

DIVISION:-SE-9

BATCH A

The screenshot shows a web browser window with the URL `onlinegdb.com/online_c_compiler`. The browser's address bar and tabs are visible. Below the browser window is a toolbar for the online compiler, including buttons for Run, Debug, Stop, Share, Save, Beautify, and a download icon. The language is set to C. The main editor area displays a C program named `main.c` with line numbers 32 to 62. The code implements a function to find the minimum absolute difference between an element in an array and its adjacent elements. It uses a `for` loop to iterate through the array, and nested `if` and `for` loops to calculate the absolute difference and update the `TotalHeadMoment` variable. The code is as follows:

```
32     int temp;
33     temp=RQ[j];
34     RQ[j]=RQ[j+1];
35     RQ[j+1]=temp;
36 }
37
38 }
39 }
40
41 int index;
42 for(i=0;i<n;i++)
43 {
44     if(initial<RQ[i])
45     {
46         index=i;
47         break;
48     }
49 }
50
51 // if movement is towards high value
52 if(move==1)
53 {
54     for(i=index;i<n;i++)
55     {
56         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
57         initial=RQ[i];
58     }
59
60     for(i=index-1;i>=0;i--)
61     {
62         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
```

NAME:-PRIYANK JHAVERI

ROLL NO:-18

DIVISION:-SE-9

BATCH A

```
63         initial=RQ[i];
64     }
65 }
66 // if movement is towards low value
67 else
68 {
69     for(i=index-1;i>=0;i--)
70     {
71         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
72         initial=RQ[i];
73     }
74     for(i=index;i<n;i++)
75     {
76         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
77         initial=RQ[i];
78     }
79 }
80 }
81 }
82 }
83
84 printf("Total head movement is %d",TotalHeadMoment);
85 return 0;
86 }
```

input

```
Enter the number of Requests
8
Enter the Requests sequence
95 180 34 119 11 123 62 64
Enter initial head position
50
Enter total disk size
200
Enter the head movement direction for high 1 and for low 0
1
Total head movement is 299

...Program finished with exit code 0
Press ENTER to exit console.
```

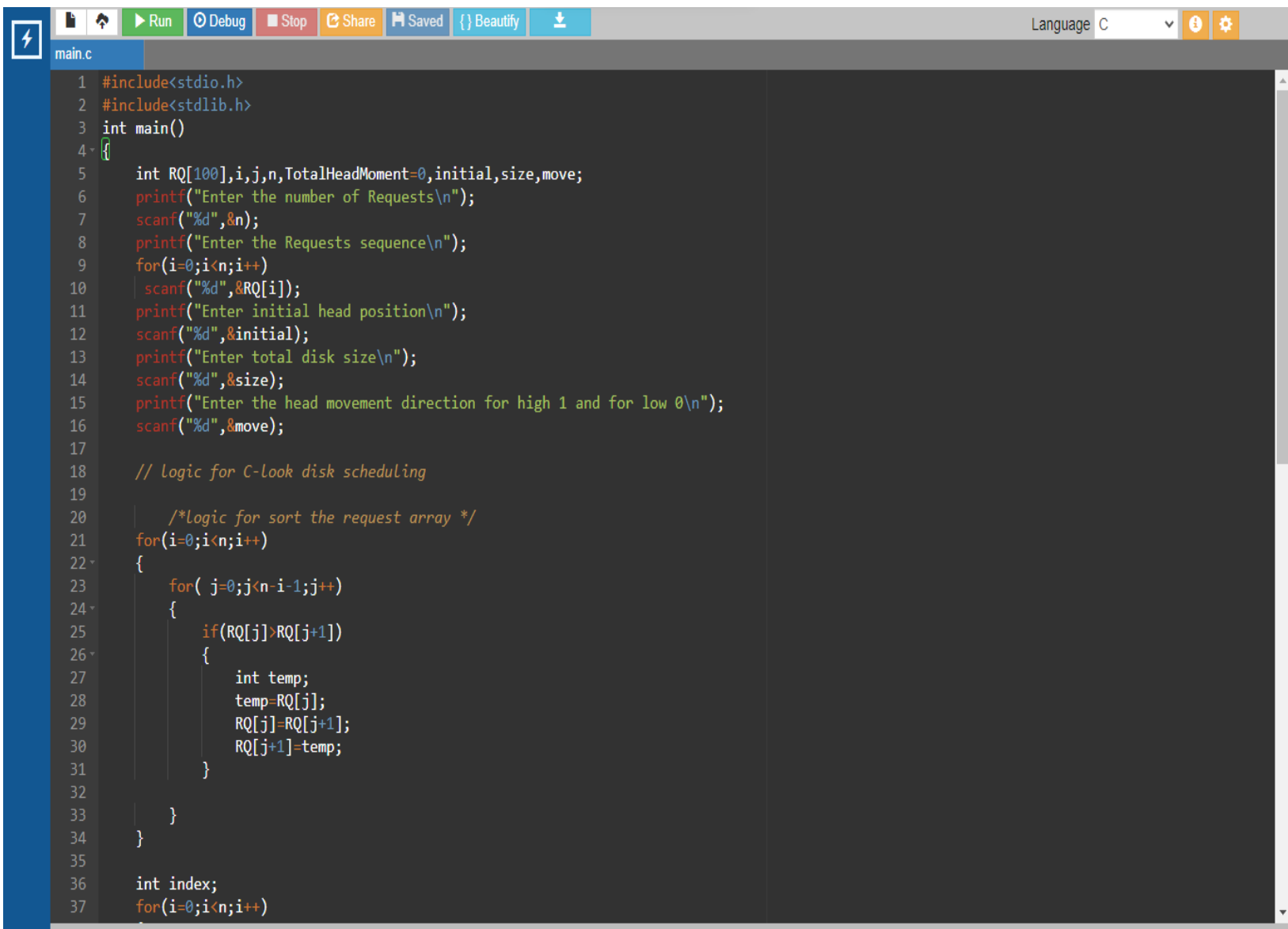
NAME:-PRIYANK JHAVERI

ROLL NO:-18

DIVISION:-SE-9

BATCH A

C-LOOK SCHEDULING



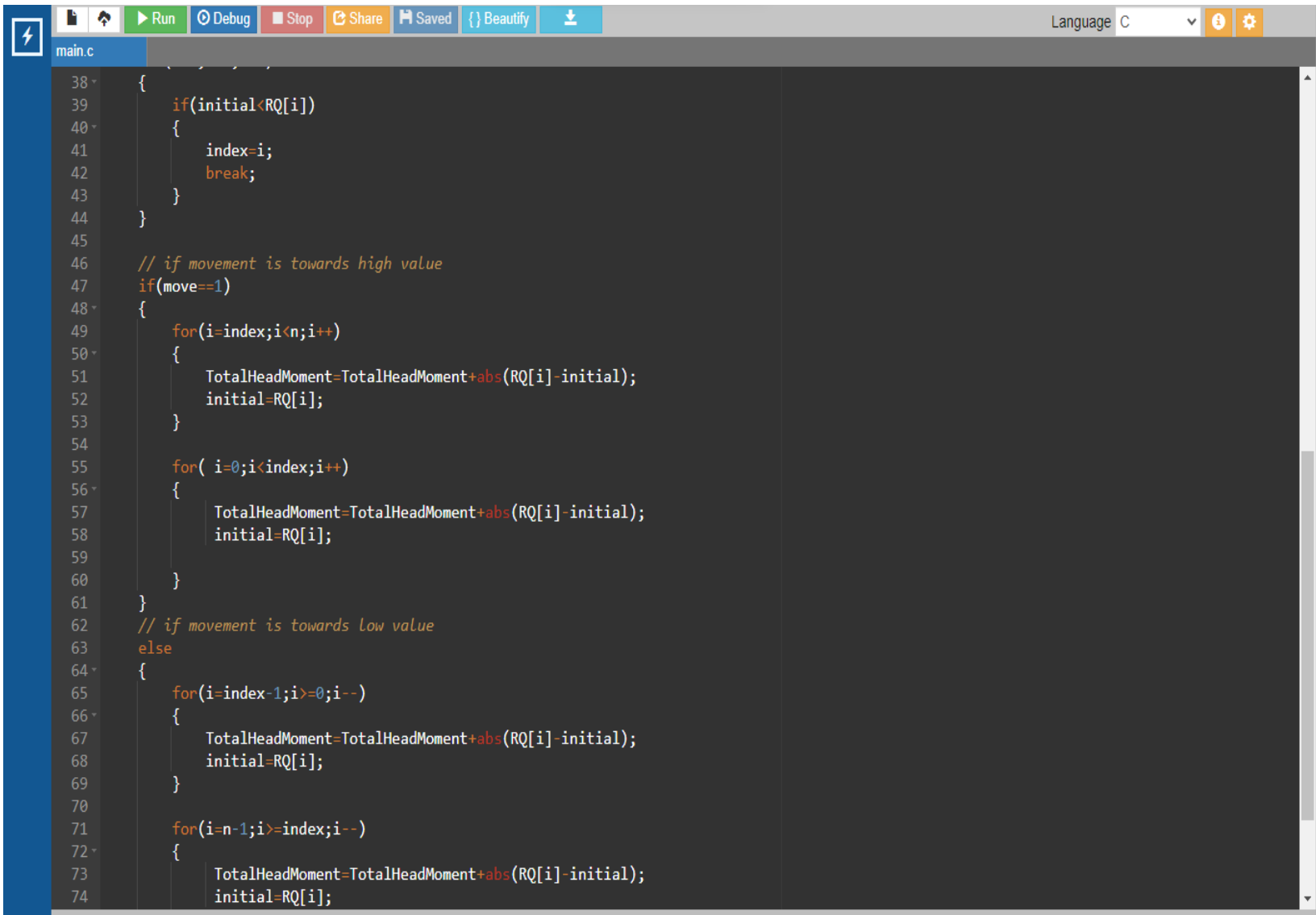
```
1 #include<stdio.h>
2 #include<stdlib.h>
3 int main()
4 {
5     int RQ[100],i,j,n,TotalHeadMoment=0,initial,size,move;
6     printf("Enter the number of Requests\n");
7     scanf("%d",&n);
8     printf("Enter the Requests sequence\n");
9     for(i=0;i<n;i++)
10         scanf("%d",&RQ[i]);
11     printf("Enter initial head position\n");
12     scanf("%d",&initial);
13     printf("Enter total disk size\n");
14     scanf("%d",&size);
15     printf("Enter the head movement direction for high 1 and for low 0\n");
16     scanf("%d",&move);
17
18     // Logic for C-Look disk scheduling
19
20     /*Logic for sort the request array */
21     for(i=0;i<n;i++)
22     {
23         for( j=0;j<n-i-1;j++)
24         {
25             if(RQ[j]>RQ[j+1])
26             {
27                 int temp;
28                 temp=RQ[j];
29                 RQ[j]=RQ[j+1];
30                 RQ[j+1]=temp;
31             }
32         }
33     }
34
35     int index;
36     for(i=0;i<n;i++)
37     {
```

NAME:-PRIYANK JHAVERI

ROLL NO:-18

DIVISION:-SE-9

BATCH A



```
main.c
38 {
39     if(initial<RQ[i])
40     {
41         index=i;
42         break;
43     }
44 }
45
46 // if movement is towards high value
47 if(move==1)
48 {
49     for(i=index;i<n;i++)
50     {
51         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
52         initial=RQ[i];
53     }
54
55     for( i=0;i<index;i++)
56     {
57         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
58         initial=RQ[i];
59     }
60 }
61
62 // if movement is towards low value
63 else
64 {
65     for(i=index-1;i>=0;i--)
66     {
67         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
68         initial=RQ[i];
69     }
70
71     for(i=n-1;i>=index;i--)
72     {
73         TotalHeadMoment=TotalHeadMoment+abs(RQ[i]-initial);
74         initial=RQ[i];
```

```
75
76     }
77 }
78
79 printf("Total head movement is %d",TotalHeadMoment);
80 return 0;
81 }
```

NAME:-PRIYANK JHAVERI

ROLL NO:-18

DIVISION:-SE-9

BATCH A



```
input
Enter the number of Requests
8
Enter the Requests sequence
95 180 34 119 11 123 62 64
Enter initial head position
50
Enter total disk size
200
Enter the head movement direction for high 1 and for low 0
1
Total head movement is 322

...Program finished with exit code 0
Press ENTER to exit console.
```