

C:\Users\khush\Documents\Untitled1.exe

Enter the maximum range of Disk: 99
Enter the number of queue requests: 7
Enter the initial head position: 24
Enter the disk positions to be read(queue): 12

26

24

4

42

8

50

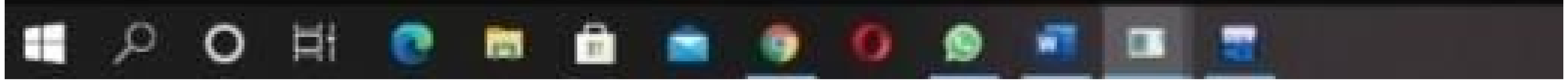
Disk head moves from position 24 to 26 with Seek 2
Disk head moves from position 26 to 42 with Seek 16
Disk head moves from position 42 to 50 with Seek 8
Disk head moves from position 50 to 99 with Seek 49
Disk head moves from position 99 to 24 with Seek 75
Disk head moves from position 24 to 12 with Seek 12
Disk head moves from position 12 to 8 with Seek 4
Disk head moves from position 8 to 4 with Seek 4

Total Seek Time= 170

Average Seek Time= 24.285715

Process exited after 88.03 seconds with return value 0

Press any key to continue . . .



Name - Rishi Wison
Name - Khushboo Upadhyay.
U-Roll No - 200230188
S-ID - 20052088
Course - BSCIT 2 'A'

Ques 2 #include <stdio.h>

int absolute value (int);

int main()

{ int queue[25], n, head position, i, j, k, seek=0, maxrange,
difference, temp, queue1[20], queue2[20], temp1=0, temp2=0;

float average Seek Time;

printf("Enter the max. range of Disk: ");

scanf("%d", &maxrange);

printf("Enter the no. of queue requests: ");

scanf("%d", &n);

printf("Enter the initial head position: ");

scanf("%d", &headposition);

printf("Enter the disk position to be read(queue): ");

for (i=1; i<=n; i++)

{ scanf("%d", &temp);

if (temp > headposition)

{ queue1[temp] = temp;

temp2++;

}

for (i=0; i<temp1-1; i++)

{ for (j=i+1; j<temp; j++)

{ if (queue1[i] > queue1[j])

{ temp = queue1[i];

khushboo


```

queue1[i] = queue1[j];
queue1[j] = temp;
}
}
}
for (j = i + 1; j < temp2; j++)
{
    if (queue2[i] < queue2[j])
    {
        temp = queue2[i];
        queue2[i] = queue2[j];
        queue2[j] = temp;
    }
}
for (i = 1; j < temp1; i++, j++)
{
    queue[i] = queue1[j];
}
queue[i] = maxrange;
for (i = temp1 + 2; j = 0; j < temp2; i++, j++)
{
    queue[i] = queue2[j];
}
queue[i] = 0;
queue[0] = headposition;
for (j = 0; j <= n, j++)
{
    difference = absoluteValue(queue[j + 1] - queue[j]);
    seek = seek + difference;
}
average Seek Time = seek / (float) n;
printf("Total Seek Time = %.d\n", seek);
printf("Average Seek Time = %.f\n",
average Seek Time);
}

```

Khus 13/03/20

int absolute value (int x)

{

if (x > 0)

{ return x;

}

else

{

return x * -1;

}

}

Khushboo
20052088
BSCIT 21A)