Name- Bhavna
Course- BSc IT section A
Std Id - 20052005

Q1.

```
~~#~~ #include <stdio.h>
~~#include <conio.h>~~
#define max 25
void main ()
{
 int frag[max], b[max], f[max], i, j, nb, nf,
   temp, highest =0;
 static int bf[max], ff[max];
printf ("\n\t Memory Scheme - Worst fit");
printf (" \n Enter the number of blocks:");
scanf (" %d", &nb);
printf ("Enter number of files:");
scanf (" %d", &nf);
printf (" \n Enter the size of the blocks:-\n");
for (i=1; i<=nb; i++)
 {
   printf ("Block %d:", i);
   scanf (" %d", & b[i]);
 }
```

REDMI NOTE 8
BHAVNA 입니다💜💜

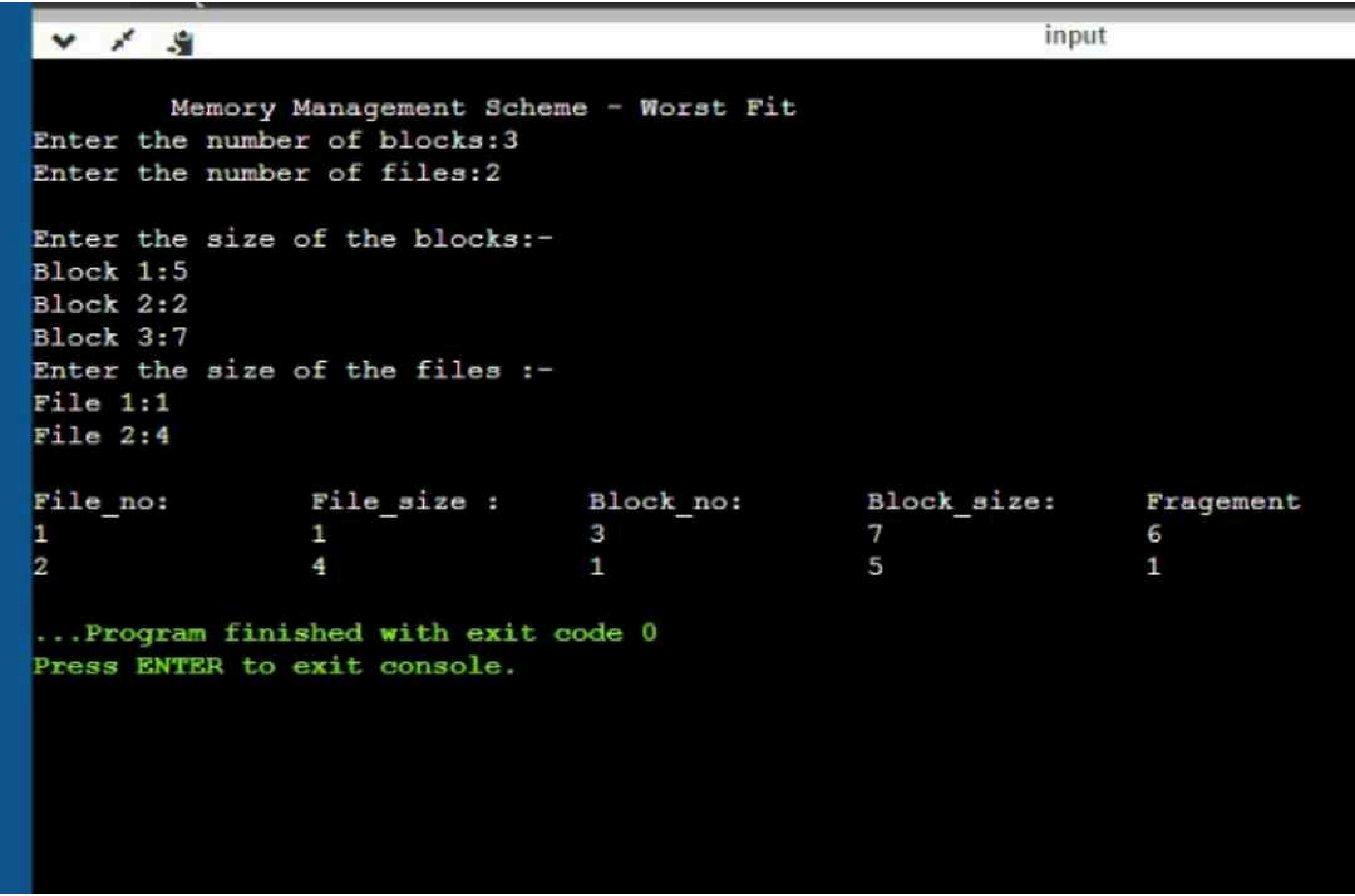
2021/8/27 15:07

```
printf(" Enter size of the files :-\n");
for(i = 1; i<=nf; i++)
 {
   printf(" File %d:", i);
   Scanf("%d", &f[i]);
  }
for (i=1; i<=nf; i++)
  {
   for(j = 1; j<=nb; j++)
    { if (bf[j]!=1)
      {
       temp = b[j]-f[i];
       if(temp >= 0)
       if(highest < temp)
        {
         ff[i]=j;
         highest = temp ;
        }
       }
     }
   frag[i] = highest;
   bf[ff[i]] = 1;
   highest = 0 ;
  }
```

REDMI NOTE 8
BHAVNA 입니다💜💜
2021/8/27 15:07

```
printf("\n file no:\t file size : \t Blockno: \t Block
  size: \t fragement");
  for (i=1; i<=nf; i++)
  printf("\n %d \t\t %d\t \t %d\t\t %d \t\t%d",
     i, f[i], ff[i], b[ff[i]], frag[i]);
  }
```

REDMI NOTE 8
BHAVNA 입니다💜💜
2021/8/27 15:07



```
        Memory Management Scheme - Worst Fit
Enter the number of blocks:3
Enter the number of files:2

Enter the size of the blocks:-
Block 1:5
Block 2:2
Block 3:7
Enter the size of the files :-
File 1:1
File 2:4

File_no:        File_size :     Block_no:       Block_size:     Fragement
1               1               3               7               6
2               4               1               5               1

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

Q2.

```
 # include <stdio.h>
 int absoluteValue (int)
 void main ()
{ int  queue [25], n, head position, i, j, K, seek =0,
  maxrange, difference, temp, queue1[20], queue2[20],
 temp1 =0, temp2 =0;
 float averageSeekTime ;
printf ("Enter the maximum range of Disk: ");
scanf(" %d", & maxrange);
printf ("Enter the number of queue requests : ");
 scanf (" %d", & n);
 printf (" Enter the initial head position:");
scanf (" %d", &headposition);
printf ( "Enter the disk positions to be read (queue):
for(i= 1; i<= n; i++)
 { scanf(" %d", &temp );
   if (temp > headposition)
     {
      queue1[temp1] = temp;
      temp1++;
     }
     ~~else~~ else
     { queue2[temp2] = temp;
       temp2++;
     }
 }
```

REDMI NOTE 8
BHAVNA 입니다💜💜

2021/8/27 15:07

```
for(i=0;i<temp1-1 ;i++)
{
  for(j=i+1 ; j<temp1; j++)
  {
    if(queue1[i]>queue 1[j])
    {
      temp =queue1[i];
      queue1[i]=queue1[j];
      queue1[ j] = temp;
    }
  }
}

for(i=0; i<temp2-1; i++)
{
   ~~for(j=i~~;
   for(j=i+1; j<temp2; j++
   {  if(queue2[i]<queue2[j])
      { temp =queue2[i];
        queue2[i]=queue2[j];
        queue2(j)=temp;
      }
   }
}
```

REDMI NOTE 8
BHAVNA 입니다💜💜

2021/8/27 15:07

```
~~for (i=1; j=0~~;
for (i=1, j=0; j< temp 1; i++, j++)
   {
      queue[i] = queue1[j];
   }
  queue[i] = max range;
 for (i= temp1+2, j=0; j<temp 2; 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;
 }
averageSeekTime = seek / (float)n;
printf(" Total Seek Time = %d \n", seek);
printf (" Average Seek Time= %f \n", averageseek
                                        time);
}
```

REDMI NOTE 8
BHAVNA 입니다💜💜

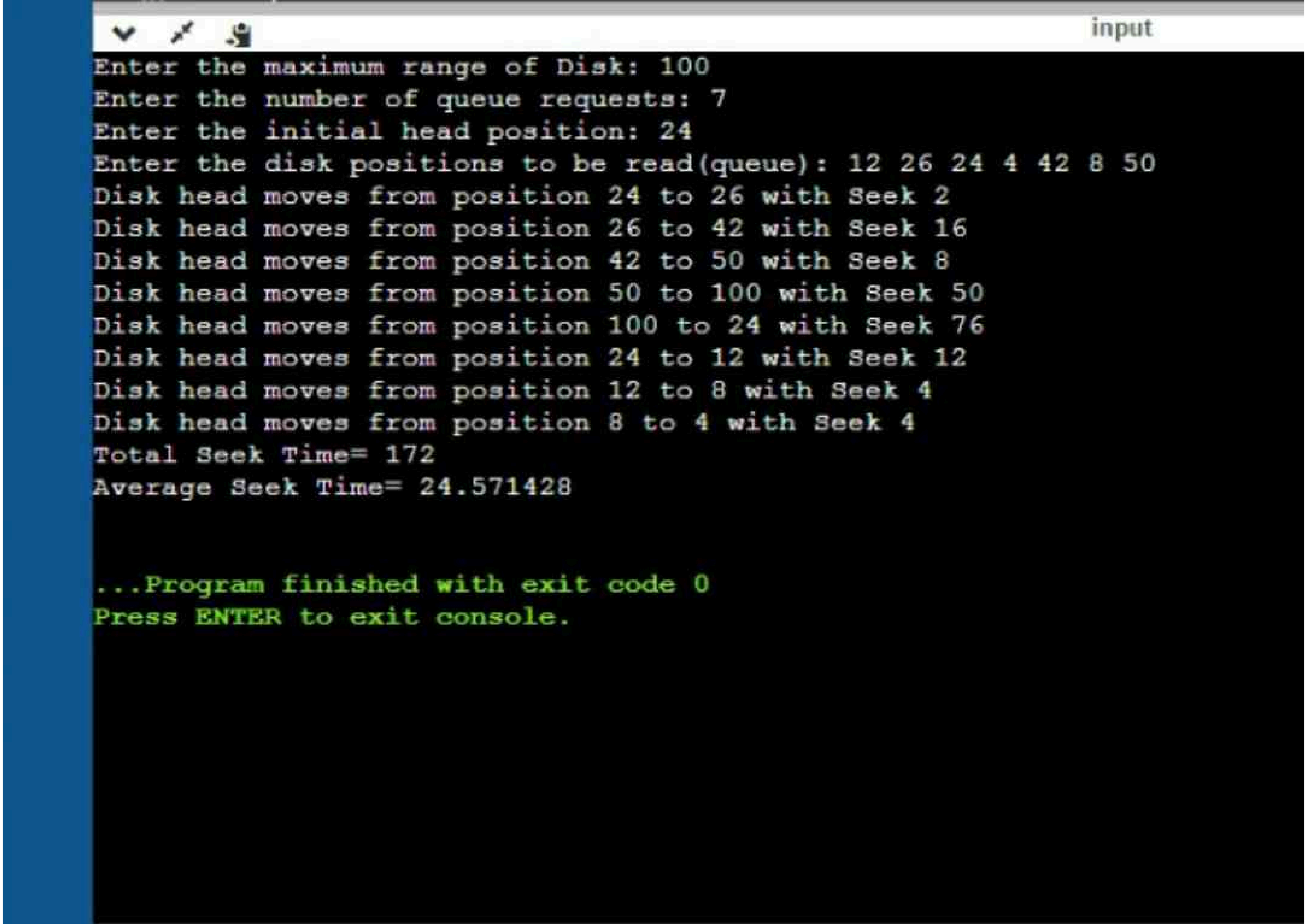
2021/8/27 15:07

~~int absolute value~~

```
int absoluteValue(int n)
{
    if(n>0)
    {
        return n;
    }
    else
    {
        return n*-1;
    }
};
```

27 Aug 2021

REDMI NOTE 8
BHAVNA 입니다💜💜
2021/8/27 15:07



```
Enter the maximum range of Disk: 100
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 100 with Seek 50
Disk head moves from position 100 to 24 with Seek 76
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= 172
Average Seek Time= 24.571428


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