NAME : ANUSHREE BAJAJ

ROLL NO : 33107

```
//program for DISK SCHEDULING ALGORITHM
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int choice,track,no_req,head,head1,distance;
int disc_req[100],finish[100];
void menu()
{
   printf("\n\n********MENU********");
   printf("\n1. Input data\n 2. SSTF \n 3. SCAN \n 4. C-LOOK \n 5. Exit");
   printf("\n\n Enter your choice");
   scanf("%d",&choice);
}
void input()
{
   int i;
   printf("Enter Total number of tracks");
   scanf("%d",&track);
   printf("Enter total number of disc requests");
   scanf("%d",&no_req);
   printf("\n Enter disc requests in FCFS order");
   for(i=0;i<no_req;i++)
   {
      scanf("%d",&disc_req[i]);

   }
   printf("\n Enter current head position");
   scanf("%d",&head1);
```

```
}

void sstf()
{
  int min,diff;
  int pending=no_req;
  int i,distance=0,index;
  head=head1;

  for(i=0;i<no_req;i++)
  {
    finish[i]=0;
  }

  printf("\n%d=>",head);

  while(pending>0)
  {   min=9999;

    for(i=0;i<no_req;i++)
    {
      diff=abs(head-disc_req[i]);
      if(finish[i]==0 && diff<min)
      {
        min=diff;
        index=i;
      }
    }
    finish[index]=1;
    distance+=abs(head-disc_req[index]);
    head=disc_req[index];
```

```
        pending--;
        printf("%d=>",head);
    }
    printf("End");
    printf("\n\n Total Distance Traversed=%d",distance);
}

void sort()
{
    int i,j,temp;
    for(i=0;i<no_req;i++)
    {
        for(j=0;j<no_req;j++)
        {
            if(disc_req[i]<disc_req[j])
            {
                temp=disc_req[i];
                disc_req[i]=disc_req[j];
                disc_req[j]=temp;
            }
        }
    }
}

void scan()
{
    int index,dir;
    int i;
    distance=0;
    head=head1;
```

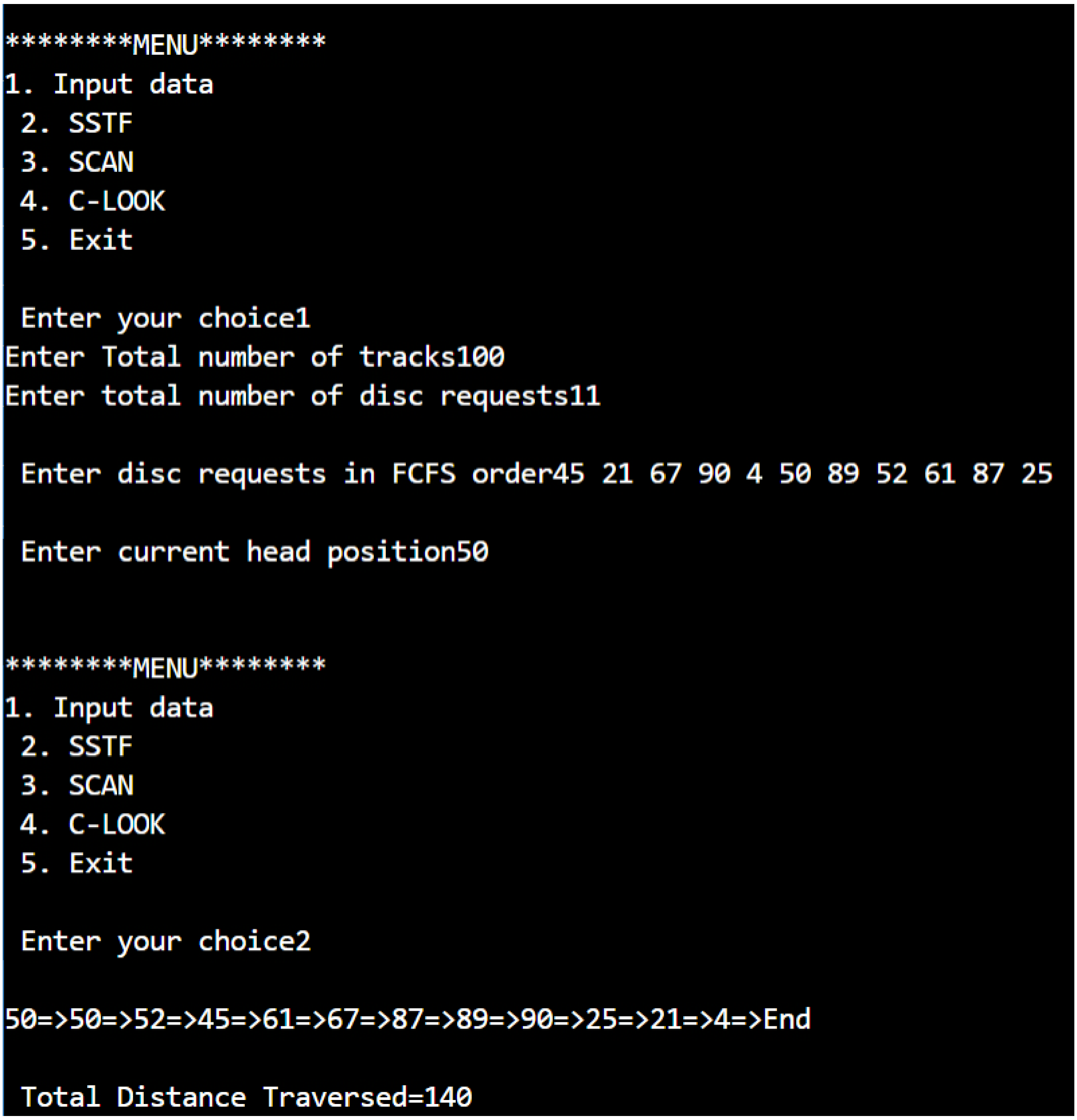
```
  printf("\n Enter the direction of head \n 1 - Towars higher disc(Right) \n 0 -towards lower disc(left)");
  scanf("%d",&dir);
  sort();
  printf("\n Sorted Disc requests are: ");
  for(i=0;i<no_req;i++)
  {

    printf("  %d",disc_req[i]);
  }

  i=0;
  while(head>=disc_req[i])
  {
    index=i;
    i++;
  }
  printf("\n index=%d",index);
  printf("\n%d=>",head);
  if(dir==1)
  {
    sort();
    for(i=index+1;i<no_req;i++)
    {
      printf("%d=>",disc_req[i]);
      distance+=abs(head-disc_req[i]);
      head=disc_req[i];
    }
    distance+=abs(head-(track-1));
    printf("%d=>",track-1);
    head=track-1;
```

```
        for(i=index;i>=0;i--)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }
    }
    else
    {
        sort();
        for(i=index;i>=0;i--)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }
        distance+=abs(head-0);
        head=0;
        printf("0=>");
        for(i=index+1;i<no_req;i++)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }

    }
    printf("End");
    printf("\n Total Distance Traversed=%d",distance);
}
```

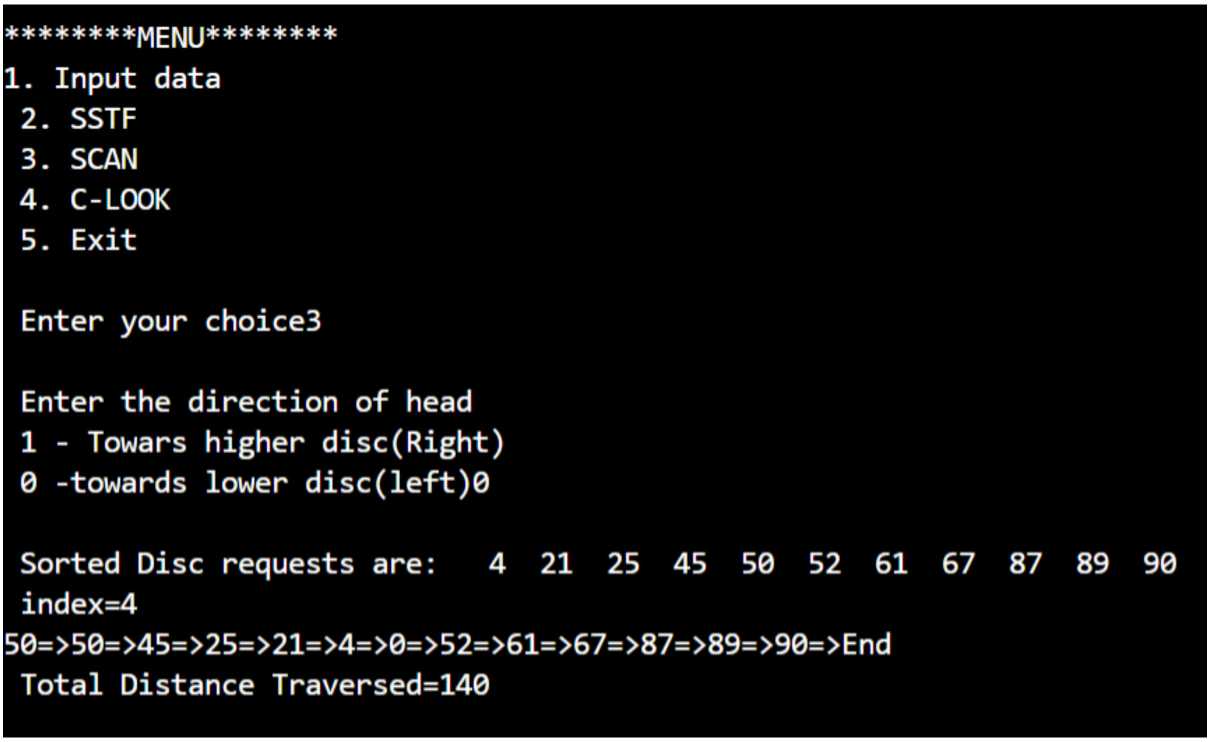
```
void clook()
{
    int index,dir;
    int i;
    distance=0;
    head=head1;
    printf("\n Enter the direction of head \n 1 - Towars higher disc \n 0 -towards lower disc");
    scanf("%d",&dir);
    sort();
    printf("\n Sorted Disc requests are: ");
    for(i=0;i<no_req;i++)
    {

        printf("  %d",disc_req[i]);
    }

    i=0;
    while(head>=disc_req[i])
    {
        index=i;
        i++;
    }
    printf("\n index=%d",index);
    printf("\n%d=>",head);
    if(dir==1)
    {
        sort();
        for(i=index+1;i<no_req;i++)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }
        for(i=0;i<index;i++)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }
    }
    else
    {
        sort();
        for(i=index;i>=0;i--)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }
        for(i=(no_req-1);i>index;i--)
        {
            printf("%d=>",disc_req[i]);
            distance+=abs(head-disc_req[i]);
            head=disc_req[i];
        }

    }
    printf("End");
    printf("\n Total Distance Traversed=%d",distance);
}

int main()
```

```
{
   while(1)
   {
      menu();
      switch(choice)
      {
      case 1: input();
         break;
      case 2: sstf();
         break;
      case 3: scan();
         break;
      case 4: clook();
         break;
      case 5: exit(0);
         break;
      default:
         printf("\n Enter valid choice");
         break;
      }
   }

   return 0;
}
```

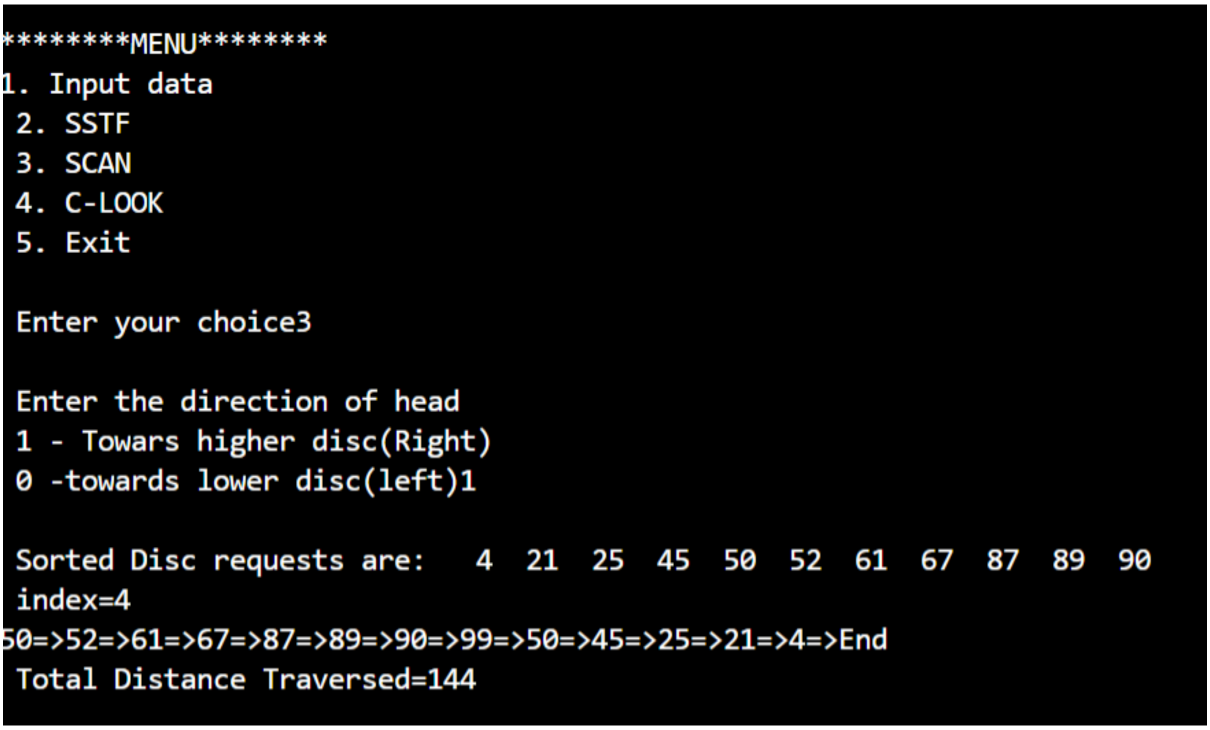
OUTPUT :



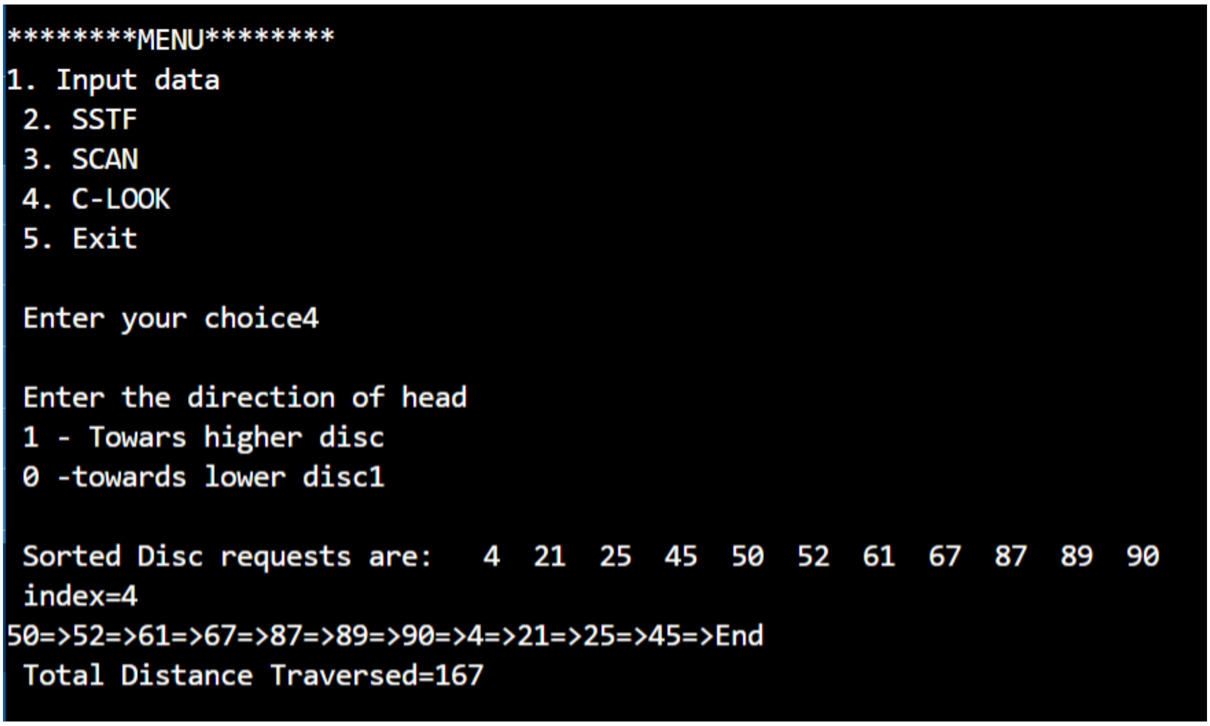
```
********MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice1
Enter Total number of tracks100
Enter total number of disc requests11

 Enter disc requests in FCFS order45 21 67 90 4 50 89 52 61 87 25

 Enter current head position50


********MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice2

50=>50=>52=>45=>61=>67=>87=>89=>90=>25=>21=>4=>End

 Total Distance Traversed=140
```



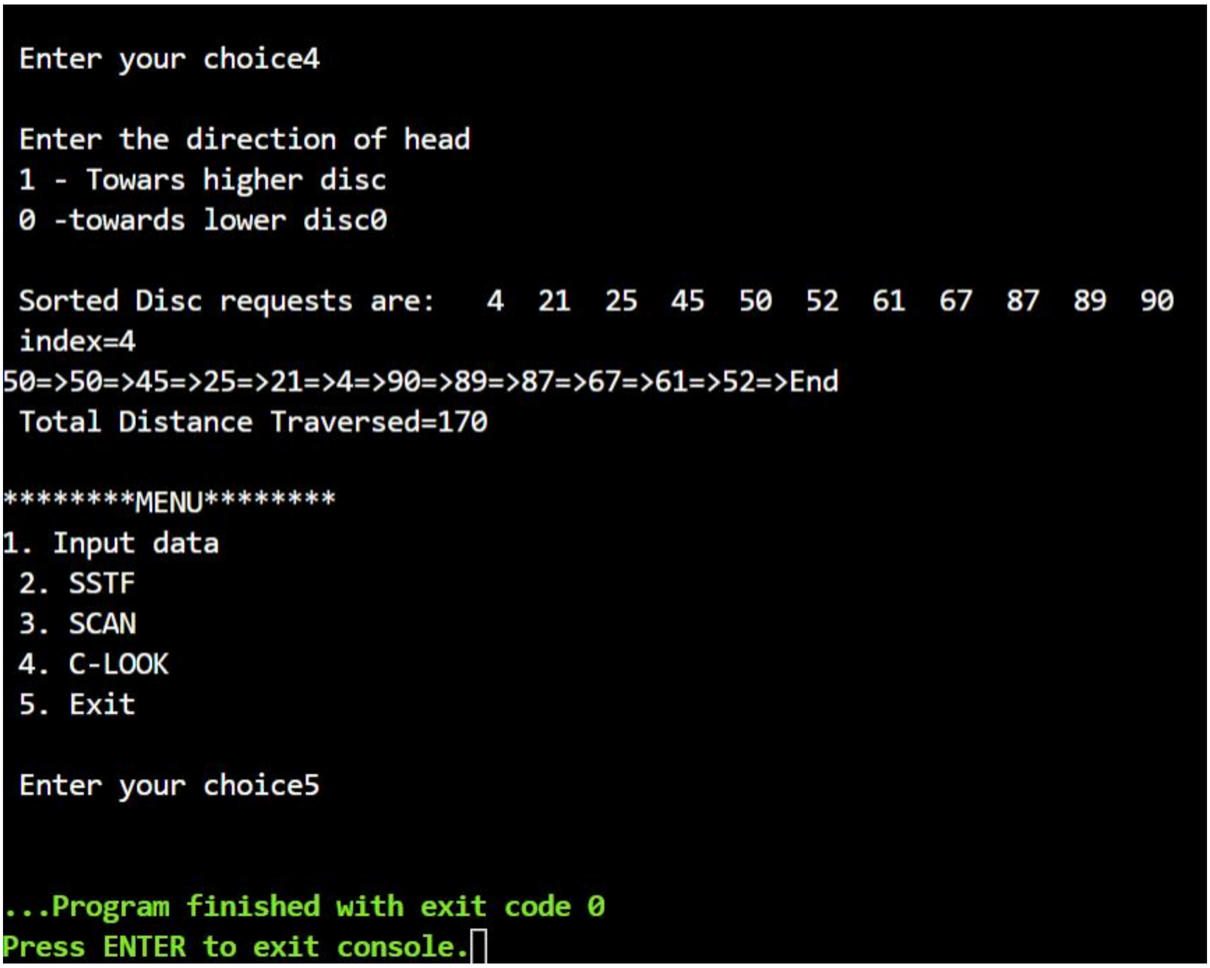
```
********MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice3

 Enter the direction of head
 1 - Towars higher disc(Right)
 0 -towards lower disc(left)0

 Sorted Disc requests are:   4   21   25   45   50   52   61   67   87   89   90
 index=4
50=>50=>45=>25=>21=>4=>0=>52=>61=>67=>87=>89=>90=>End
 Total Distance Traversed=140
```



```
********MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice3

 Enter the direction of head
 1 - Towars higher disc(Right)
 0 -towards lower disc(left)1

 Sorted Disc requests are:   4   21   25   45   50   52   61   67   87   89   90
 index=4
50=>52=>61=>67=>87=>89=>90=>99=>50=>45=>25=>21=>4=>End
 Total Distance Traversed=144
```



```
*******MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice4

 Enter the direction of head
 1 - Towars higher disc
 0 -towards lower disc1

 Sorted Disc requests are:   4   21   25   45   50   52   61   67   87   89   90
 index=4
50=>52=>61=>67=>87=>89=>90=>4=>21=>25=>45=>End
 Total Distance Traversed=167
```



```

 Enter your choice4

 Enter the direction of head
 1 - Towars higher disc
 0 -towards lower disc0

 Sorted Disc requests are:   4   21   25   45   50   52   61   67   87   89   90
 index=4
50=>50=>45=>25=>21=>4=>90=>89=>87=>67=>61=>52=>End
 Total Distance Traversed=170

*******MENU********
1. Input data
 2. SSTF
 3. SCAN
 4. C-LOOK
 5. Exit

 Enter your choice5


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