Experiment No:10

Experiment Name: FCFS Disk Scheduling Algorithm.

Aim and Objectives:

All incoming requests are placed at the end of the queue. Whatever number that is next in the queue will be the next number served. Using this algorithm doesn't provide the best results. To determine the number of head movements you would simply find the number of tracks it took to move from one request to the next.

Code:

#include<stdio.h>

#include<conio.h>

void main()

{

int queue[100],n,head,i,j,k,seek=0,diff;

float avg;

// clrscr();

printf("\*\*\* FCFS Disk Scheduling Algorithm \*\*\*\n");

printf("Enter the size of Queue\t");

scanf("%d",&n);

printf("Enter the Queue\t");

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

{

scanf("%d",&queue[i]);

}

printf("Enter the initial head position\t");

scanf("%d",&head);

queue[0]=head;

printf("\n");

for(j=0;j<=n-1;j++)

{

diff=abs(queue[j+1]-queue[j]);

seek+=diff;

printf("Move from %d to %d with Seek %d\n",queue[j],queue[j+1],diff);

}

printf("\nTotal Seek Time is %d\t",seek);

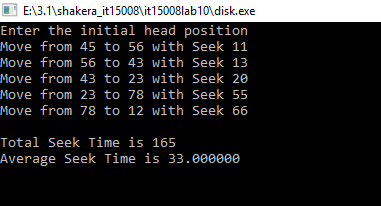
avg=seek/(float)n;

printf("\nAverage Seek Time is %f\t",avg);

getch();

}

Output:



Discussion:

In this lab we learn about fcfs disk scheduling. It is the simplest form of disk scheduling algorithms. The I/O requests are served or processes according to their arrival. The request arrives first will be accessed and served first. Since it follows the order of arrival, it causes the wild swings from the innermost to the outermost tracks of the disk and vice versa.