

WEEK 10

Write a C program to simulate disk scheduling algorithms

a) FCFS

b) SCAN

c) C-SCAN

CODE:

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
int disks;
```

```
void quicksort(int number[25], int first, int last)
```

```
{
```

```
    int i, j, pivot, temp;
```

```
    if (first < last)
```

```
    {
```

```
        pivot = first;
```

```
        i = first;
```

```
        j = last;
```

```
        while (i < j)
```

```
        {
```

```
            while (number[i] <= number[pivot] && i < last)
```

```
                i++;
```

```
            while (number[j] > number[pivot])
```

```
                j--;
```

```
            if (i < j)
```

```
            {
```

```
                temp = number[i];
```

```
                number[i] = number[j];
```

```
                number[j] = temp;
```

```
            }
```

```

    }
    temp = number[pivot];
    number[pivot] = number[j];
    number[j] = temp;
    quicksort(number, first, j - 1);
    quicksort(number, j + 1, last);
}
}
void fcfs(int arr[],int src, int n)
{
    int sseq[20],i;

    sseq[0]=abs(arr[0]-src);
    for(i=1;i<n;i++)
        sseq[i]=abs(arr[i]-arr[i-1]);

    int sum=0;
    for(i=0;i<n;i++)
        sum+=sseq[i];

    printf("\nFCFS \nTotal seek sequenece: %d \nSeek Sequence: \n",sum);
    for(i=0;i<n;i++)
        printf("%d ",sseq[i]);
    printf("\n");
}

void cscan(int arr[], int src, int n)
{
    int i,sum=0,j,sseq[20];
    quicksort(arr, 0, n-1);
    int index;
    for (index = 0; index < n; index++) {
        if (arr[index] == src) {
            break;
        }
    }

```

```

}
i=index+1;
j=0;
while(i<=n)
{
    sseq[j]=abs(arr[i]-arr[i-1]);
    i++;
    j++;
}
sseq[j++]=abs(disks-arr[i-1]);
i=0;
sseq[j++]=abs(disks);
sseq[j++]=abs(arr[0]);
while(i<index)
{
    sseq[j++]=abs(arr[i]-arr[i-1]);
    i++;
}
for(i=0;i<(n+2);i++)
sum+=sseq[i];

```

```

    printf("\nC-SCAN \nTotal seek sequence: %d \nSeek Sequence:
\n",sum);
    for(i=0;i<n+2;i++)
        printf("%d ",sseq[i]);
    printf("\n");

```

```

}
void scan(int arr[], int src, int n)
{
    int i,sum=0,j,sseq[20];
    quicksort(arr, 0, n-1);
    int index;
    for (index = 0; index < n; index++) {
        if (arr[index] == src) {

```

```

        break;
    }
}
i=index+1;
j=0;
while(i<=n)
{
    sseq[j]=abs(arr[i]-arr[i-1]);
    i++;
    j++;
}
sseq[j++]=abs(disks-arr[i-1]);

i=index-1;
sseq[j++]=abs(arr[i]-disks);
while(i>=0)
{
    sseq[j++]=abs(arr[i]-arr[i-1]);
    i--;
}
for(i=0;i<(n+2);i++)
sum+=sseq[i];

printf("\nSCAN \nTotal seek sequenece: %d \nSeek Sequence: \n",sum);
for(i=0;i<n+2;i++)
printf("%d ",sseq[i]);
printf("\n");
}
void main()
{
    int source, arr[20],i,n,copy[20];
    printf("Enter number of disks: ");
    scanf("%d",&n);

    printf("\nEnter %d values: ",n);

```

```

for(i=0;i<n;i++)
scanf("%d",&arr[i]);

printf("\nEnter source position: ");
scanf("%d",&source);

printf("\nEnter number disks: ");
scanf("%d",&disks);

for(i=0;i<n;i++)
copy[i]=arr[i];

arr[n]=source;
copy[n]=arr[n];

fcfs(copy , source , n);
scan(copy , source , n);
cscan(arr , source , n);
}

```

OUTPUT:

```

"C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\OS\bin\Debug\OS.exe"
Enter number of disks: 5
Enter 5 values: 10 25 30 45 12
Enter source position: 19
Enter number disks: 50
FCFS
Total seek sequence: 77
Seek Sequence:
9 15 5 15 33
SCAN
Total seek sequence: 81
Seek Sequence:
31 5 15 5 13 2 10
C-SCAN
Total seek sequence: 116
Seek Sequence:
31 50 10 5 2 13 5

```