Pgm.No.3

DISK SCHEDULING ALGORITHMS

AIM

Simulate the following disk scheduling algorithms
a). FCFS
b). SCAN
c). C-SCAN

FIRST COME FIRST SERVE (FCFS)

PROGRAM

```
#include<stdio.h>
void main(){

	int ioq[20],i,n,ihead,tot;
	float seek=0,avgs;

	printf("Enter the number of requests\t:");
	scanf("%d",&n);
	printf("Enter the initial head position\t:");
	scanf("%d",&ihead);
	ioq[0] = ihead;
	ioq[n+1] =0;

	printf("Enter the I/O queue requests \n");
	for(i=1;i<=n;i++){
		scanf("%d",&ioq[i]);
	}
	ioq[n+1] =ioq[n];// to set the last seek zero

	printf("\nOrder of request served\n");
	for(i=0;i<=n;i++){

		tot = ioq[i+1] - ioq[i];
		if(tot < 0)
			tot = tot * -1;
		seek += tot;
		// printf("%d\t%d\n",ioq[i],tot);// to display each seek
		printf("%d --> ",ioq[i]);

	}
```

```
        avgs = seek/(n);

        printf("\nTotal Seek time\t\t: %.2f",seek);
        printf("\nAverage seek time\t: %.2f\n\n",avgs);
}
```

OUTPUT 1

```
Enter the number of requests  :5
Enter the initial head position :100
Enter the I/O queue requests
23
89
132
42
187

Order of request served
100 --> 23 --> 89 --> 132 --> 42 --> 187 -->
Total Seek time          : 421.00
Average seek time        : 84.20
```

OUTPUT 2

```
Enter the number of requests  :5
Enter the initial head position :100
Enter the I/O queue requests
23
89
132
42
187

Order of request served
100      77
23       66
89       43
132      90
42       145
187      0

Total Seek time          : 421.00
Average seek time        : 84.20
```

SCAN

PROGRAM

```
#include<stdio.h>
void main()
{
	int ioq[20],i,n,j,ihead,temp,scan,tot;
	float seek=0,avgs;

	printf("Enter the number of requests\t:");
	scanf("%d",&n);
	printf("Enter the initial head position\t:");
	scanf("%d",&ihead);
	ioq[0] = ihead;
	ioq[1] = 0;
	n += 2;
	printf("Enter the I/O queue requests \n");
	for(i=2;i<n;i++){
		scanf("%d",&ioq[i]);
	}

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

			if(ioq[j] > ioq[j+1]){

				temp = ioq[j];
				ioq[j] = ioq[j+1];
				ioq[j+1] = temp;

			}

		}
	}
	ioq[n]=ioq[n-1];
	for(i=0;i<n;i++){

		if(ihead == ioq[i]){

			scan = i;
			break;
```

```
        }

    }

    printf("\nOrder of request served\n\n");
    tot = 0;
    for(i=scan;i>=0;i--){
        //rai  tot = ioq[i+1] - ioq[i];
        tot = ioq[i] – ioq[i-1]; // me
        if(i==0) // me
            tot=ioq[i]-ioq[scan+1];//me
        if(tot < 0)
            tot = tot * -1;
        //seek += tot;
        printf("%d\t%d\n",ioq[i],tot);
    }


    for(i=scan+1;i<n;i++){
        tot = ioq[i+1] - ioq[i];
        if(tot < 0)
            tot = tot * -1;
        //seek += tot;
        printf("%d\t%d\n",ioq[i],tot);
    }
    seek = ihead + ioq[n-1];

    avgs = seek/(n-2);

    printf("\n\nTotal Seek time\t\t: %.2f",seek);
    printf("\nAverage seek time\t: %.2f\n\n",avgs);

}
```

OUTPUT

```
Enter the number of requests  :8
Enter the initial head position :53
Enter the I/O queue requests
98
183
37
122
14
```

124
65
67

Order of request served

53	16
37	23
14	14
0	65
65	2
67	31
98	24
122	2
124	59
183	0

Total Seek time : 236.00
Average seek time : 29.50

CSCAN

PROGRAM

```
#include<stdio.h>
void main()
{
	int ioq[20],i,n,j,ihead,itail,temp,scan,tot=0;
	float seek=0,avgs;

	printf("Enter the number of requests\t: ");
	scanf("%d",&n);
	ioq[0] = 0;
	printf("Enter the initial head position\t: ");
	scanf("%d",&ihead);
	ioq[1] = ihead;
	printf("Enter the maximum track limit\t: ");
	scanf("%d",&itail);
	ioq[2] = itail;
	n += 3;

	printf("Enter the I/O queue requests \n");
	for(i=3;i<n;i++){
```

```
        scanf("%d",&ioq[i]);
}


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

                if(ioq[j] > ioq[j+1]){

                        temp = ioq[j];
                        ioq[j] = ioq[j+1];
                        ioq[j+1] = temp;

                }

        }
}

for(i=0;i<n+1;i++){

        if(ihead == ioq[i]){

                scan = i;
                break;

        }

}

i = scan;
temp = n;

printf("\nOrder of request served\n");
printf("\n");

while(i != temp){

        if(i < temp-1){
                tot = ioq[i+1] - ioq[i];

                if(tot < 0)
                        tot = tot * -1;
                seek += tot;
        }
        printf("%d --> ",ioq[i]);
        // printf("%d\t%d\n",ioq[i],tot);
        i++;

        if(i == n){
```

```
                i = 0;
                temp = scan;
                seek += itail;

            }

        }

        avgs = seek/(n-3);

        printf("\n\nTotal Seek time\t\t: %.2f",seek);
        printf("\nAverage seek time\t: %.2f\n\n",avgs);
}
```

OUTPUT

```
Enter the number of requests  : 8
Enter the initial head position : 50
Enter the maximum track limit          : 200
Enter the I/O queue requests
90
120
35
122
38
128
65
68

Order of request served

50 --> 65 --> 68 --> 90 --> 120 --> 122 --> 128 --> 200 --> 0 --> 35 --> 38 -->

Total Seek time          : 388.00
Average seek time      : 48.50
```