A.Y.: 2023-24 Sem: III Sub: Operating Systems Laboratory

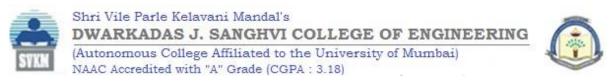
Name: Anish Sharma Div: I-1 Roll no: I011 SAP ID: 60003220045

# **Experiment 7**

Aim: Implement Sequential File allocation techniques

#### Code:

```
#include <stdio.h>
#include <stdlib.h>
int flag[100];
void main()
{
  int f[50], n, i, st, len, j, c, k, count = 0;
  for (i = 0; i < 50; i++)
     f[i] = 0;
  printf("Enter no. of files:");
  scanf("%d", &n);
  count = 0;
  printf("Enter memory req:");
  int mem[n];
  for (int i = 0; i < n; i++)
     scanf("%d", &mem[i]);
  for (int i = 0; i < n; i++)
     count = 0;
     int index = rand() \% 100;
     int copy = index;
     if (flag[index] == 1)
       i--;
       continue;
     for (int j = 0; j < mem[i]; j++)
       flag[index] = 1;
       index++;
       count++;
     printf("\nfile:%d,length%d block allocated are:", i + 1, count);
     for (int j = index; j < index + count; j++)
       printf("%d", j);
  }
```



A.Y.: 2023-24 Sem: III Sub: Operating Systems Laboratory }
Output:
Enter no. of files: 5
5
Enter memory req : 4 5 6 4 5
4 5 6 4 5

file:1,length4 block allocated are:87 88 89 90
file:2,length5 block allocated are:82 83 84 85 86
file:3,length6 block allocated are:21 22 23 24 25 26
file:4,length4 block allocated are:97 98 99 100
file:5,length5 block allocated are:40 41 42 43 44

A.Y.: 2023-24 Sem: III Sub: Operating Systems Laboratory

#### **EXPERIMENT 8**

Aim: Implement FCFS, SSTF, SCAN, CSCAN disk scheduling algorithm.(Any 2)

#### 1)FCFS

```
Code:
import java.util.*;
public class OS1 {
  public static void main(String[] args) {
    Scanner sc=new Scanner(System.in);
    int queue[]=\{98, 183, 40, 122, 10, 124, 65\};
    int head=sc.nextInt();
    int temp=head;
    int FCFS=0;
    int min=Integer.MAX VALUE;
    System.out.println("The order of the pointer movement in FCFS is:");
    for(int i=0;i<queue.length;i++){
       System.out.print(queue[i]+"->");
       FCFS=FCFS+Math.abs(temp-queue[i]);
       temp=queue[i];
    System.out.println("\n total seek time with FCFS algorithm is:"+FCFS);
Output:
 PS D:\College assignments\OS> java OS1
 The order of the pointer movement in FCFS is:
 98-> 183-> 40-> 122-> 10-> 124-> 65->
```

## 2) SSTF

#### Code:

```
import java.util.*;
public class SSTF {
   public static void main(String[] args) {
      Scanner sc=new Scanner(System.in);
      int head=sc.nextInt();
      int temp=head;
      int SSTF=0;
```

total seek time with FCFS algorithm is:640

```
A.Y.: 2023-24
                                     Sem: III
                                                            Sub: Operating Systems Laboratory
    int min=Integer.MAX_VALUE;
    int minIndex=-1;
    System.out.println("input values for SSTF:");
    Vector<Integer> v=new Vector<Integer>();
    for(int j=0; j<7; j++){
       v.add(sc.nextInt());
    System.out.println("The order of the pointer movement in SSTF is:");
    temp=head;
    for(int x=0; x<7; x++){
       for(int y=0;y< v.size();y++){
         if(Math.abs(v.elementAt(y)-temp)<min){
           min=Math.abs(v.elementAt(y)-temp);
           minIndex=y;
       System.out.print(v.elementAt(minIndex)+"-> ");
       SSTF=SSTF+min;
       temp=v.elementAt(minIndex);
       v.removeElement(v.elementAt(minIndex));
       min=Integer.MAX_VALUE;
    System.out.println("\n total seek time with SSTF algorithm is:"+SSTF);
  }
Output:
PS D:\College assignments\OS> java SSTF
input values for SSTF:
98 183 40 122 10 124 65
The order of the pointer movement in SSTF is:
40-> 65-> 98-> 122-> 124-> 183-> 10->
 total seek time with SSTF algorithm is:326
```

## 3) SCAN

#### Code:

```
import java.util.*;
public class SCAN {
   public static void main(String[] args) {
      Scanner sc=new Scanner(System.in);
      int queue[]={98, 183, 40, 122, 10, 124, 65};
      int head=sc.nextInt();
      int temp=-1,temp2=head;
      int SCAN=0;
      Arrays.sort(queue);
      for(int j=0;j<queue.length;j++){</pre>
```

```
A.Y.: 2023-24
                                     Sem: III
                                                            Sub: Operating Systems Laboratory
       if(queue[j]>head){
         temp=j;
         break;
       }
     }
    System.out.println("The order of the pointer movement in SCAN is:");
    for(int i=temp-1;i>=0;i--){
       System.out.print(queue[i]+"->");
       SCAN=SCAN+(temp2-queue[i]);
       temp2=queue[i];
    SCAN=SCAN+temp2;
    temp2=0;
    for(int k=temp; k<7; k++){
       System.out.print(queue[k]+"->");
       SCAN=SCAN+(Math.abs(temp2-queue[k]));
       temp2=queue[k];
    System.out.println("\n total seek time with SCAN algorithm is:"+SCAN);
Output:
PS D:\College assignments\OS> java SCAN
 The order of the pointer movement in SCAN is:
 40-> 10-> 65-> 98-> 122-> 124-> 183->
 total seek time with SCAN algorithm is:233
4) C-SCAN
Code:
import java.util.*;
public class CSCAN {
  public static void main(String[] args) {
    Scanner sc=new Scanner(System.in);
    int queue[]={98, 183, 40, 122, 10, 124, 65};
    int head=sc.nextInt();
    int temp=-1,temp2=head;
    int SCAN=0;
    Arrays.sort(queue);
    for(int j=0;j<queue.length;j++){
       if(queue[j]>head){
         temp=i;
         break;
    System.out.println("The order of the pointer movement in C-SCAN is:");
    for(int i=temp-1;i>=0;i--){
```

```
A.Y.: 2023-24
                                    Sem: III
                                                           Sub: Operating Systems Laboratory
       System.out.print(queue[i]+"->");
       SCAN=SCAN+(temp2-queue[i]);
       temp2=queue[i];
    for(int k=temp; k<7; k++){
       System.out.print(queue[k]+"->");
       SCAN=SCAN+(Math.abs(temp2-queue[k]));
       temp2=queue[k];
    System.out.println("\n total seek time with C-SCAN algorithm is:"+SCAN);
Output:
 PS D:\College assignments\OS> java CSCAN
 The order of the pointer movement in C-SCAN is:
 40-> 10-> 65-> 98-> 122-> 124-> 183->
  total seek time with C-SCAN algorithm is:213
5) LOOK
Code:
```

```
import java.util.*;
public class LOOK {
  public static void main(String[] args) {
    Scanner sc=new Scanner(System.in);
    int queue[]={98, 183, 40, 122, 10, 124, 65};
    int head=sc.nextInt();
    int temp=-1,temp2=head;
    int LOOK=0;
    Arrays.sort(queue);
    for(int j=0;j<queue.length;j++){
       if(queue[i]>head){
         temp=j;
         break;
       }
     }
    System.out.println("The order of the pointer movement in LOOK is:");
    for(int i=temp-1;i>=0;i--){
       System.out.print(queue[i]+"->");
       LOOK=LOOK+(temp2-queue[i]);
       temp2=queue[i];
    System.out.print("0 ->");
    LOOK=LOOK+temp2+199;
```

```
A.Y.: 2023-24
                                      Sem: III
                                                             Sub: Operating Systems Laboratory
    temp2=199;
    System.out.print("199 ->");
    for(int k=6;k>=temp;k--){
       System.out.print(queue[k]+"->");
       LOOK=LOOK+(Math.abs(temp2-queue[k]));
       temp2=queue[k];
    System.out.println("\n total seek time with LOOK algorithm is:"+LOOK);
  }
}
Output:
 PS D:\College assignments\OS> java LOOK
 The order of the pointer movement in LOOK is:
 40-> 10-> 0 ->199 ->183-> 124-> 122-> 98-> 65->
 total seek time with LOOK algorithm is:383
6) C-LOOK
Code:
import java.util.*;
public class CLOOK {
  public static void main(String[] args) {
    Scanner sc=new Scanner(System.in);
    int queue[]={98, 183, 40, 122, 10, 124, 65};
    int head=sc.nextInt();
    int temp=-1,temp2=head;
    int LOOK=0;
    Arrays.sort(queue);
    for(int j=0;j<queue.length;j++){</pre>
       if(queue[j]>head){
         temp=j;
         break;
     }
    System.out.println("The order of the pointer movement in C-LOOK is:");
    for(int i=\text{temp-1}; i>=0; i--){
       System.out.print(queue[i]+"->");
```

LOOK=LOOK+(temp2-queue[i]);

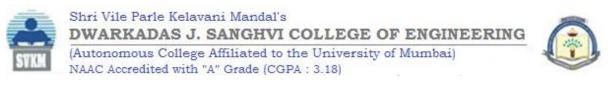
System.out.print(queue[k]+"->");

LOOK=LOOK+queue[6]-temp2;

temp2=queue[i];

for(int k=6;k>=temp;k--)

temp2=199;



```
A.Y.: 2023-24 Sem: III Sub: Operating Systems Laboratory LOOK=LOOK+(Math.abs(temp2-queue[k])); temp2=queue[k]; }
System.out.println("\n total seek time with C-LOOK algorithm is:"+LOOK); }
Output:

PS D:\College assignments\OS> java CLOOK
50
The order of the pointer movement in C-LOOK is: 40-> 10-> 183-> 124-> 122-> 98-> 65-> total seek time with C-LOOK algorithm is:347
```