



DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING

**Title: Simulating the Sequential, Linked and
Indexed File Allocation Strategies.**

OPERATING SYSTEM LAB
CSE 310



GREEN UNIVERSITY OF BANGLADESH

1 Objective(s)

- To gather knowledge of different file allocation strategies.
- To implement Sequential, Linked and Indexed file allocation strategies.

2 Problem analysis

A file is a collection of data, usually stored on disk. As a logical entity, a file enables to divide data into meaningful groups. As a physical entity, a file should be considered in terms of its organization. The term "file organization" refers to the way in which data is stored in a file and, consequently, the method(s) by which it can be accessed. In this section, Sequential file allocation strategy is discussed.

2.1 Sequential File Allocation Strategy

In this file organization, the records of the file are stored one after another both physically and logically. That is, record with sequence number 16 is located just after the 15th record. A record of a sequential file can only be accessed by reading all the previous records.

3 Algorithm

Algorithm 1: Sequential File Allocation

- 1 Step 1: Start the program.
 - 2 Step 2: Get the number of memory partition and their sizes.
 - 3 Step 3: Get the number of processes and values of block size for each process.
 - 4 Step 4: First fit algorithm searches all the entire memory block until a hole which is big enough is encountered. It allocates that memory block for the requesting process.
 - 5 Step 5: Best-fit algorithm searches the memory blocks for the smallest hole which can be allocated to requesting process and allocates it.
 - 6 Step 6: Worst fit algorithm searches the memory blocks for the largest hole and allocates it to the process.
 - 7 Step 7: Analyses all the three memory management techniques and display the best algorithm which utilizes the memory resources effectively and efficiently.
 - 8 Step 8: Stop the program.
-

4 Implementation in C

```
1  /* Sequential File Allocation Strategy code */
2  #include<stdio.h>
3  #include<conio.h>
4
5  int main()
6  {
7      int f[50], i, st, len, j, c, k, count = 0;
8
9      for(i=0;i<50;i++)
10         f[i]=0;
11
12     printf("  Files Allocated are : \n");
13
14     x: count=0;
15
16     printf("  Enter starting block and length of files: ");
17     scanf("%d %d", &st,&len);
18
```

```

19  for(k=st;k<(st+len);k++)
20      if(f[k]==0)
21          count++;
22
23  if(len==count)
24  {
25      for(j=st;j<(st+len);j++)
26          if(f[j]==0)
27          {
28              f[j]=1;
29              printf("   %d\t%d\n",j,f[j]);
30          }
31
32      if(j!=(st+len-1))
33          printf("   The file is allocated to disk\n");
34  }
35  else
36      printf("   The file is not allocated \n");
37
38  printf("   Do you want to enter more file(Yes - 1/No - 0)");
39  scanf("%d", &c);
40
41  if(c==1)
42      goto x;
43
44  return 0;
45  }

```

5 Input/Output

Output of the Sequential File Allocation Strategy program is given below.

```

Files Allocated are :
Enter starting block and length of files: 4 5
4      1
5      1
6      1
7      1
8      1
The file is allocated to disk
Do you want to enter more file(Yes - 1/No - 0)1
Enter starting block and length of files: 2 1
2      1
The file is allocated to disk
Do you want to enter more file(Yes - 1/No - 0)1
Enter starting block and length of files: 9 5
9      1
10     1
11     1
12     1
13     1
The file is allocated to disk
Do you want to enter more file(Yes - 1/No - 0)0
-----

```

6 Discussion & Conclusion

Based on the focused objective(s) to understand about Sequential File Allocation Strategy, the additional lab exercises made me more confident towards the fulfilment of the objectives(s).

7 Lab Task (Please implement yourself and show the output to the instructor)

1. Implement Linked File Allocation Strategy.

7.1 Problem analysis

With linked allocation, each file is a linked list of disk blocks; the disk blocks may be scattered anywhere on the disk. The directory contains a pointer to the first and last blocks of the file. Each block contains a pointer to the next block.

7.2 Algorithm of Linked File Allocation Strategy

Algorithm 2: Sequential File Allocation

- 1 Step 1: Start the program.
 - 2 Step 2: Gather information about the number of files.
 - 3 Step 3: Allocate random locations to the files.
 - 4 Step 4: Check if the location that is selected is free or not.
 - 5 Step 5: If the location is free set the flag=0 a location is allocated set the flag = 1.
 - 6 Step 6: Print the file number, length, and the block allocated.
 - 7 Step 7: Gather information if more files have to be stored.
 - 8 Step 8: If yes, then go to STEP 2.
 - 9 Step 9: If no, Stop the program.
-

7.3 Sample Output

```
Enter how many blocks already allocated: 3
Enter blocks already allocated: 1 3 5
Enter index starting block and length: 2 2
2——>1
3 Block is already allocated
4——>1
Do you want to enter more file(Yes - 1/No - 0)0
```

8 Lab Exercise (Submit as a report)

- Implement Indexed File Allocation Strategy to get the following output.

Sample Output

```
Enter the index block: 5
Enter no of blocks needed and no of files for the index 5 on the disk : 4
1 2 3 4
Allocated
File Indexed
5——>1 : 1
5——>2 : 1
5——>3 : 1
5——>4 : 1
Do you want to enter more file(Yes - 1/No - 0)1
Enter the index block: 4
```

4 index is already allocated
Enter the index block: 6
Enter no of blocks needed and no of files for the index 6 on the disk : 2
7 8
Allocated
File Indexed
6——>7 : 1
6——>8 : 1
Do you want to enter more file(Yes - 1/No - 0)0

9 Policy

Copying from the Internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.