CS331

S5 SEMESTER

System Software Lab

Github : [ceccs18c59/cs331: System Software Lab (github.com)]

Experiment No 3

Write a C program to simulate the following file allocation strategies

1. Sequential

2. Indexed

3. Linked

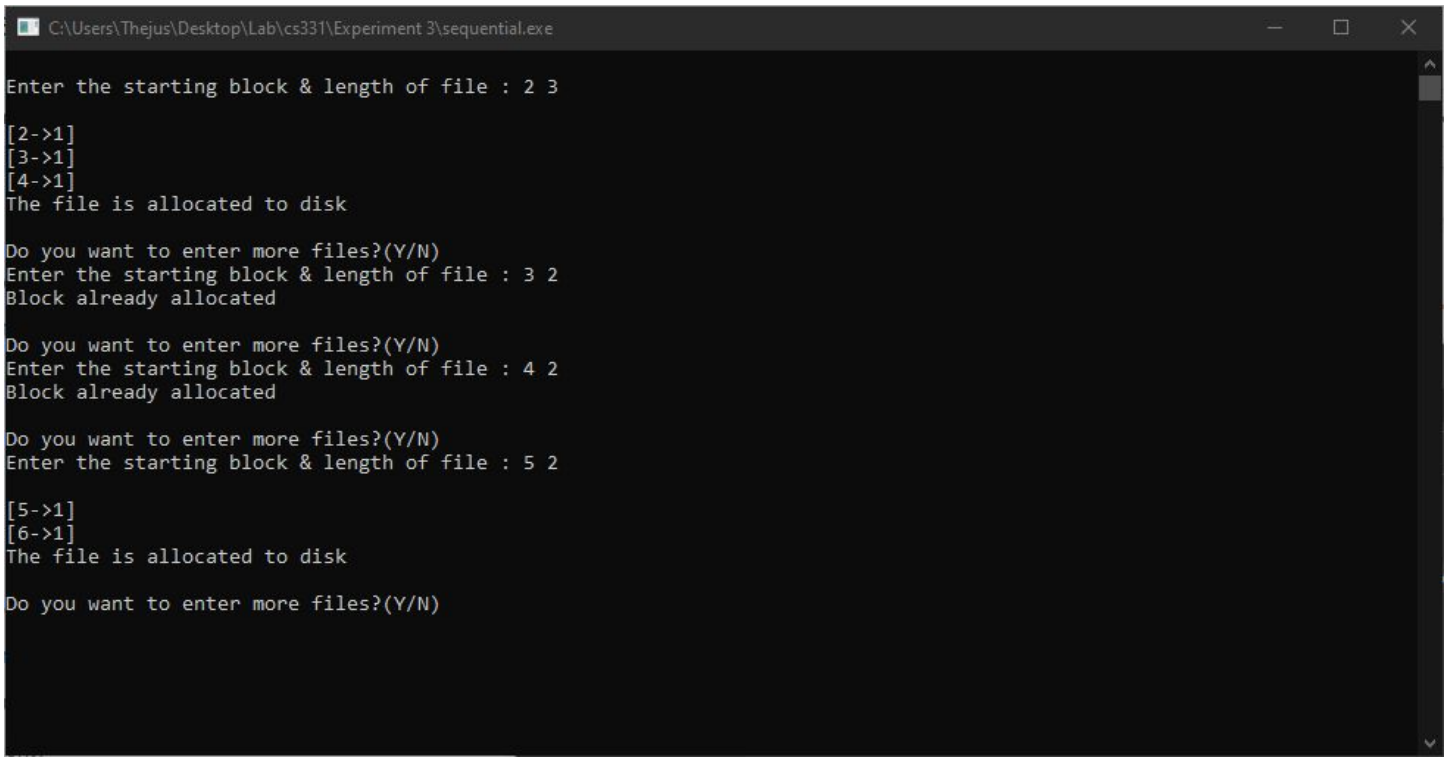
1. Sequential File Allocation

Program

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

int main()
{
    int nof;
    int memory[50];
    int status, firstBlock, blockSize, j;
    char c = 'y';

    // Initialize Memory Blocks
    for (int i = 0; i < 50; i++)
        memory[i] = 0;
    for (; c == 'Y' || c == 'y';)
    {

        printf("\nEnter the starting block & length of file : ");
        scanf("%d %d", &firstBlock, &blockSize);
        for (j = firstBlock; j < (firstBlock + blockSize); j++)
            if (memory[j] == 0)
            {
                memory[j] = 1;
                printf("\n[%d->%d]", j, memory[j]);
            }
            else
            {
                printf("Block already allocated");
```

```
                break;
            }
        if (j == (firstBlock + blockSize))
            printf("\nThe file is allocated to disk");
        printf("\n\nDo you want to enter more files?(Y/N)");
        c = getch();
    }
    getch();
    return 0;
}
```

Output



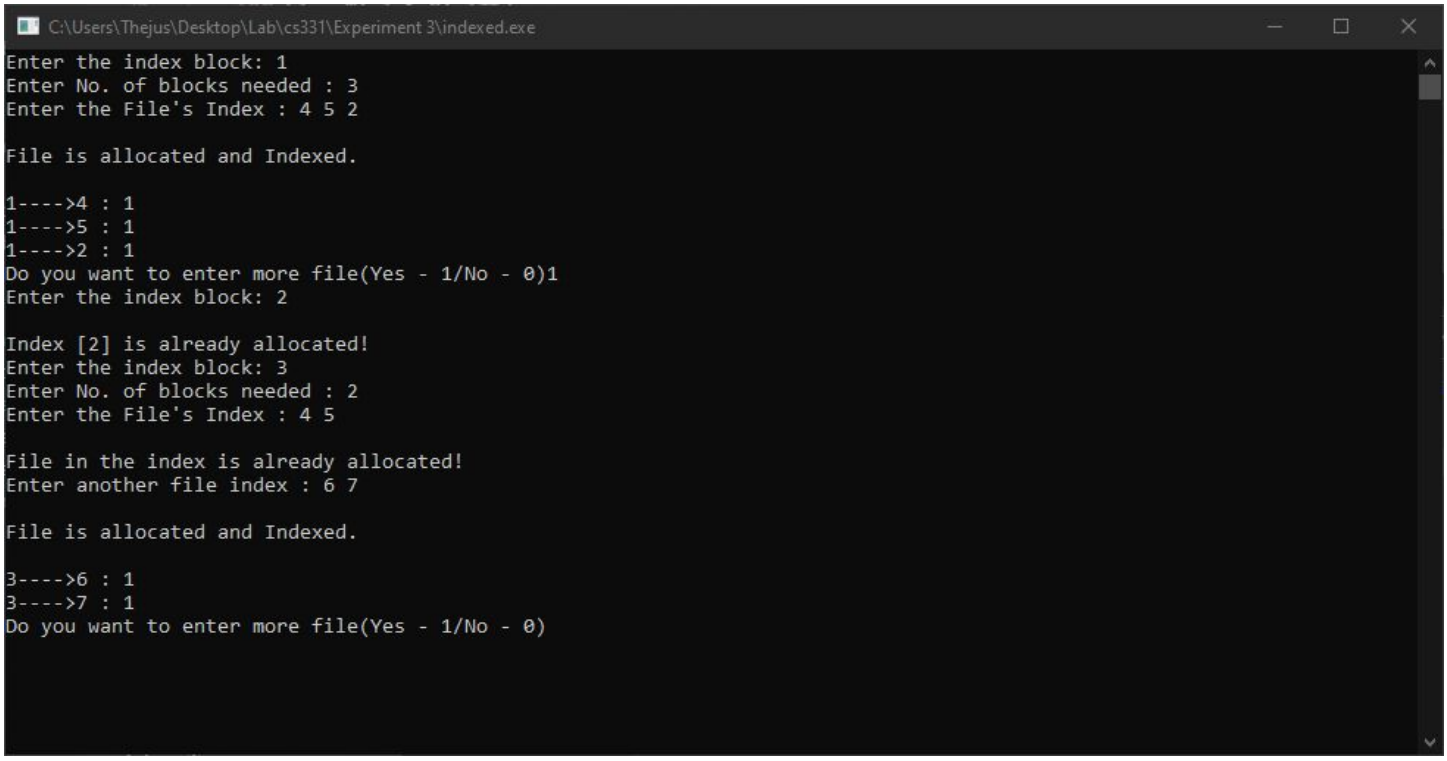
2. Indexed File Allocation

Program

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main()
{
    int f[50], index[50], i, n, st, len, j, c, k, ind, count = 0;

    for (i = 0; i < 50; i++)
        f[i] = 0;
x:
    printf("Enter the index block: ");
    scanf("%d", &ind);
    if (f[ind] != 1)
    {
        printf("Enter No. of blocks needed : ");
        scanf("%d", &n);
        printf("Enter the File's Index : ");
    }
    else
    {
        printf("\nIndex [%d] is already allocated!\n", ind);
        goto x;
    }
y:
    count = 0;
    for (i = 0; i < n; i++)
    {
        scanf("%d", &index[i]);
        if (f[index[i]] == 0)
            count++;
    }
    if (count == n)
    {
        for (j = 0; j < n; j++)
            f[index[j]] = 1;
        printf("\nFile is allocated and Indexed.\n\n");
        for (k = 0; k < n; k++)
            printf("%d---->%d : %d\n", ind, index[k], f[index[k]]);
    }
    else
    {
        printf("\nFile in the index is already allocated! \n");
        printf("Enter another file index : ");
        goto y;
    }
    printf("Do you want to enter more file(Yes - 1/No - 0)");
    scanf("%d", &c);
    if (c == 1)
        goto x;
```

```
    else
        exit(0);
    getch();
}
```

Output



3. Linked File Allocation

Program

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

void allocateFile(int file[])
{
    int st, len, k, c, j;
    printf("\nEnter the index of the starting block and its length: ");
    scanf("%d %d", &st, &len);
    k = len;
    if (file[st] == 0)
    {
        for (j = st; j < (st + k); j++)
        {
            if (file[j] == 0)
            {
                file[j] = 1;
                printf("%d------>%d\n", j, file[j]);
            }
            else
            {
                printf("The block %d is already allocated \n", j);
                k++;
            }
        }
    }
    else
        printf("The block %d is already allocated \n", st);
    printf("\nDo you want to enter more files? \n");
    printf("Enter 1 for Yes, Enter 0 for No: ");
    scanf("%d", &c);
    if (c == 1)
        allocateFile(file);
    else
        exit(0);
    return;
}

int main()
{
    int file[50], p, a;

    for (int i = 0; i < 50; i++)
        file[i] = 0;
    printf("Enter the number of blocks already allocated: ");
    scanf("%d", &p);
    printf("Enter the blocks already allocated: ");
    for (int i = 0; i < p; i++)
    {
```

```
        scanf("%d", &a);
        file[a] = 1;
    }

    allocateFile(file);
    getch();
    return 0;
}
```

Output