192311035

M.RAVI SAI VINAY

EXPERIMENT-35

Consider a file system that brings all the file pointers together into an index block. The ith entry in the index block points to the ith block of the file. Design a C program to simulate the file allocation strategy.

AIM:-

To simulate a file allocation strategy using an index block where each entry in the index block points to a specific file block.

ALGORITHM:-

1. Define an index block as an array of pointers.
2. Each pointer in the index block will point to a file block.
3. The file blocks will store data (could be integers, characters, or strings).
4. Implement functions to simulate file block allocation and access via the index block.
5. Display the contents of the file by accessing the index block and the file blocks.

PROCEDURE:-

1. Create an array to represent the file blocks.
2. Create an index block that holds pointers to each of the file blocks.
3. Implement a function to simulate the process of reading the file using the index block.
4. Use a loop to access each file block through the index block and print the content.

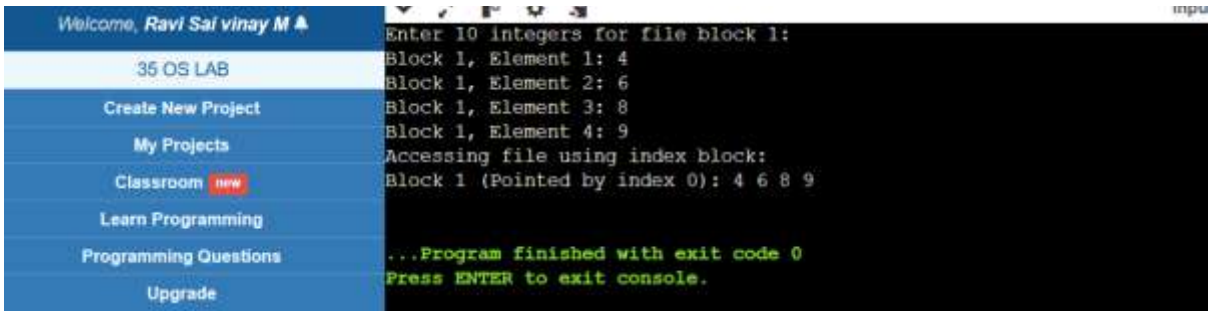
CODE:-

```
#include <stdio.h>

#include <stdlib.h>
```

```
#define MAX_BLOCKS 5  // Number of blocks in the file

// Function to simulate indexed file allocation
void accessFileThroughIndexBlock(int *indexBlock[], int fileBlocks[][10], int numBlocks) {
    printf("Accessing file using index block:\n");

    for (int i = 0; i < numBlocks; i++) {
        printf("Block %d (Pointed by index %d): ", i + 1, indexBlock[i] - fileBlocks);
        for (int j = 0; j < 10; j++) {
            printf("%d ", fileBlocks[i][j]);
        }
        printf("\n");
    }
}

int main() {
    int fileBlocks[MAX_BLOCKS][10];  // Each block can hold 10 integers
    int *indexBlock[MAX_BLOCKS];      // Index block holding pointers to file blocks

    // Simulating file data for each block
    for (int i = 0; i < MAX_BLOCKS; i++) {
        printf("Enter 10 integers for file block %d:\n", i + 1);
        for (int j = 0; j < 10; j++) {
```

```
            printf("Block %d, Element %d: ", i + 1, j + 1);
            scanf("%d", &fileBlocks[i][j]);
        }
        indexBlock[i] = fileBlocks[i];  // Pointing index block entry to the file block
    }

    // Access and display the file data using the index block
    accessFileThroughIndexBlock(indexBlock, fileBlocks, MAX_BLOCKS);

    return 0;
}
```

OUTPUT:-



RESULT:-

The program successfully simulates indexed file allocation. The index block holds pointers to each file block, and the program correctly accesses and displays the contents of each file block through the index block. The data stored in each block is shown when accessed through the index block, simulating a file system with indexed allocation.