Sequential

#include <stdio.h>

int main() {

int n, i, j;

int b[20], sb[20], t[20], c[20][20]; // Arrays:

// b[] = number of blocks for each file

// sb[] = starting block of each file

// t[] = to keep original starting block (for display)

// c[][] = to store block numbers occupied by each file

// Input: number of files

printf("Enter the number of files: ");

scanf("%d", &n);

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

// Input number of blocks the file occupies

printf("Enter the number of blocks occupied by file %d: ", i + 1);

scanf("%d", &b[i]);

// Input starting block of file

printf("Enter the starting block of file %d: ", i + 1);

scanf("%d", &sb[i]);

t[i] = sb[i]; // Store original starting block

// Fill the blocks for this file

for (j = 0; j < b[i]; j++) {

c[i][j] = sb[i]++;

}

}

// Display the starting block and length of each file

printf("\nFileName\tStart Block\tLength\n");

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

printf("%d\t\t%d\t\t%d\n", i + 1, t[i], b[i]);

}

// Display the blocks occupied by each file

printf("\nBlocks occupied are:\n");

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

printf("File no %d:", i + 1);

for (j = 0; j < b[i]; j++) {

printf("\t%d", c[i][j]);

}

printf("\n");

}

return 0;

}

START

1. Input number of files → n

2. For each file i from 0 to n-1:

a. Input number of blocks → b[i]

b. Input starting block → sb[i]

c. Store sb[i] in t[i] for display later

d. For each block j from 0 to b[i]-1:

i. Assign block number to c[i][j] = sb[i]

ii. Increment sb[i]

3. Print file number, starting block, and length

4. For each file:

a. Print all blocks occupied using c[i][j]

END