/\*

# PROGRAM TO IMPLEMENT FILE ALLOCATION STRATEGIES - SEQUENTIAL

Ashis Solomon

CS4B 17

MDL20CS035

\*/

**CODE:**

#include <stdio.h>

int files[100], start[10], len[10], alloc[10];

void allocate(int fno) {

int count = 0;

for (int i = start[fno]; i < (start[fno] + len[fno]); i++)

if (files[i] == 0)

count++;

if (count == len[fno]) {

for (int i = start[fno]; i < (start[fno] + len[fno]); i++)

files[i] = 1;

alloc[fno] = 1;

} else

alloc[fno] = 0;

}

void display(int n) {

printf("\nFile No.\tStarting block\tLength\tStatus\n");

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

if (alloc[i] == 1)

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

else

printf("%d\t\t-\t\t-\tUnallocated\n", (i + 1));

}

}

int main() {

int n;

for (int i = 0; i < 100; i++)

files[i] = 0;

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

scanf("%d", & n);

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

printf("\nEnter the starting location of the file %d: ", (i + 1));

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

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

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

allocate(i);

if (alloc[i] == 1)

printf("File %d was successfully allocated!\n", (i + 1));

else

printf("Unable to allocate disk space to File %d\n", (i + 1));

}

printf("\n-----------------------------\n");

printf("The file allocation table is: \n");

display(n);

return 0;

}

**OUTPUT:**

