**LAB TEST**

**UCS -1411 – OSLAB**

**Name :** Rahul Ram M

**Register Number :** 185001121

**Group Number :** 11

**CODE:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

/\*

Name : Rahul Ram M

Register No : 185001121

Class : CSE - 'B'

\*/

int randomAlloc(int blocks[], int num){

srand(time(0));

int count = 0;

while(count < num){

int rand\_num = (rand() % 50);

if(blocks[rand\_num] == -1){

blocks[rand\_num] = 0;

count++;

}

}

}

int divide(int num){

return (num/4 + ((num%4 != 0)?1:0));

}

void allocateFiles(int blocks[], int files[]){

int block, i, j, count, flag;

for(i = 0; i < 3; i++){

block = (divide(files[i]) > 5)?divide(files[i]):5;

count = 0;

flag = 0;

for(j = 0; j < 50; j++){

if(blocks[j] == -1){

count++;

}

else{

count = 0;

}

if(count == block){

flag = 1;

break;

}

}

if(flag == 1){

for(int k = j-count+1; k <= (j-count)+divide(files[i]); k++){

blocks[k] = i+1;

}

}

}

}

void printFiles(int blocks[], int files[]){

int flag, i, j;

for(i = 0; i < 3; i++){

flag = 0;

for(j = 0; j < 50; j++){

if(blocks[j] == i+1){

flag = 1;

break;

}

}

printf("Filename : File%d\n",i+1);

if(flag == 1){

printf("Starting block: %d\n",j+1);

printf("Length :%d\n",divide(files[i]));

}

else{

printf("Not enough free blocks!\n");

}

printf("\n");

}

}

int main(){

int blocks[50];

int files[] = {11, 17, 21};

for(int i = 0; i < 50; i++){

blocks[i] = -1;

}

randomAlloc(blocks, 17);

printf("Intermediaite result:\n\n");

printf("After assigning 17 random occupied disk blocks:\n");

printf("-1 -> free blocks\n0 -> occupied disk blocks\n\n");

for(int i = 0; i < 50; i++){

printf("%d ",blocks[i]);

}

allocateFiles(blocks, files);

printf("\n\nAfter Allocation(1-File1,2-File2,3-File3):\n\n");

for(int i = 0; i < 50; i++){

printf("%d ",blocks[i]);

}

printf("\n\n");

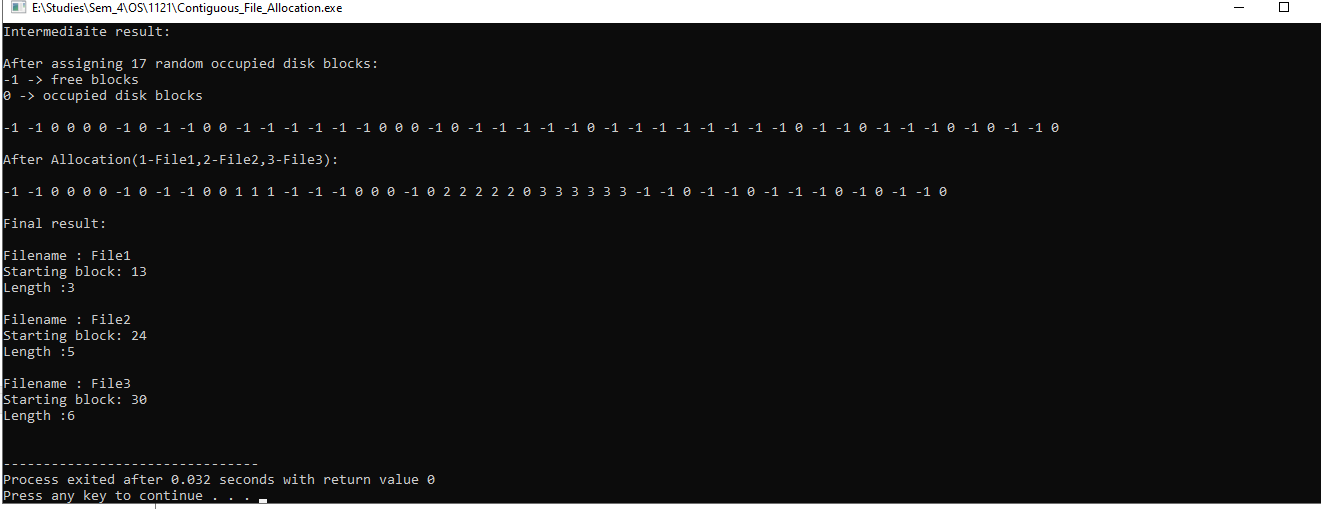
printf("Final result:\n\n");

printFiles(blocks, files);

return 0;

}

**Sample I/O:**

****