25 disk calculation

#include <stdio.h>

#define FILE\_SIZE 100

#define BLOCK\_SIZE 1

int contiguousAllocation(int blockPosition) {

if (blockPosition < 0 || blockPosition > FILE\_SIZE) {

return -1;

}

return 1;

}

int linkedAllocation(int blockPosition) {

if (blockPosition < 0 || blockPosition > FILE\_SIZE) {

return -1;

}

return blockPosition + 2;

}

int indexedAllocation(int blockPosition) {

if (blockPosition < 0 || blockPosition > FILE\_SIZE) {

return -1;

}

return 2;

}

int main() {

int blockPosition;

char choice;

printf("Enter the block position (0-%d): ", FILE\_SIZE - 1);

scanf("%d", &blockPosition);

printf("Select the allocation strategy:\n");

printf("1. Contiguous\n");

printf("2. Linked\n");

printf("3. Indexed\n");

printf("Enter your choice (1-3): ");

scanf(" %c", &choice);

int diskIO;

switch (choice) {

case '1':

diskIO = contiguousAllocation(blockPosition);

break;

case '2':

diskIO = linkedAllocation(blockPosition);

break;

case '3':

diskIO = indexedAllocation(blockPosition);

break;

default:

printf("Invalid choice!\n");

return 0;

}

if (diskIO == -1) {

printf("Invalid block position!\n");

} else {

printf("Number of disk I/O operations required: %d\n", diskIO);

}

return 0;

}

OUTPUT

