25. Write the C program to Calculate how many disk I/O operations are required for contiguous, linked, and indexed (single-level) allocation strategies, if, for one block, the following conditions hold in a file currently consisting of 100 blocks. Assume that the file control block (and the index block, in the case of indexed allocation) is already in memory.

Test Cases:

a. The block is added at the beginning.

b. The block is added in the middle.

c. The block is added at the end.

Program:-

#include <stdio.h>

#define FILE\_SIZE 100

int calculateContiguousIO(char position) {

int ioOperations = 0;

if (position == 'b') {

// Block added at the beginning

ioOperations = FILE\_SIZE + 1;

} else if (position == 'm') {

// Block added in the middle

ioOperations = FILE\_SIZE / 2 + 1;

} else if (position == 'e') {

// Block added at the end

ioOperations = FILE\_SIZE + 1;

}

return ioOperations;

}

// Function to calculate the number of disk I/O operations for linked allocation

int calculateLinkedIO(char position) {

int ioOperations = 0;

if (position == 'b' || position == 'e') {

// Block added at the beginning or end

ioOperations = FILE\_SIZE + 2;

} else if (position == 'm') {

// Block added in the middle

ioOperations = FILE\_SIZE / 2 + 3;

}

return ioOperations;

}

// Function to calculate the number of disk I/O operations for indexed (single-level) allocation

int calculateIndexedIO(char position) {

int ioOperations = 0;

if (position == 'b' || position == 'e') {

// Block added at the beginning or end

ioOperations = 2;

} else if (position == 'm') {

// Block added in the middle

ioOperations = 3;

}

return ioOperations;

}

int main() {

char position;

printf("Enter the position (b for beginning, m for middle, e for end): ");

scanf("%c", &position);

int contiguousIO = calculateContiguousIO(position);

int linkedIO = calculateLinkedIO(position);

int indexedIO = calculateIndexedIO(position);

printf("Disk I/O operations for contiguous allocation: %d\n", contiguousIO);

printf("Disk I/O operations for linked allocation: %d\n", linkedIO);

printf("Disk I/O operations for indexed (single-level) allocation: %d\n", indexedIO);

return 0;

}

Output:-

