M.HAMZA KHAN
DT-22048

LAB 14

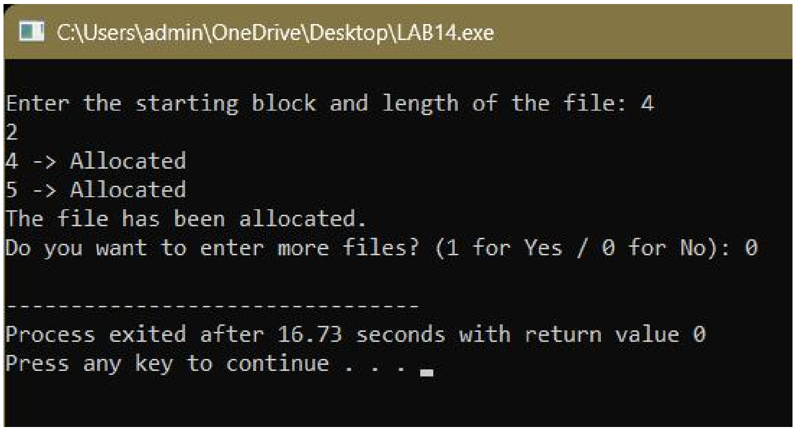
CODE:

A) SEQUENTIAL

```
#include <stdio.h>


int main() {
   int f[50] = {0}; // Block status: 0 = free, 1 = allocated
   int i, st, len, j, c;


   do {
      printf("\nEnter the starting block and length of the file: ");
      scanf("%d %d", &st, &len);


      int allocated = 1;
      for (j = st; j < st + len; j++) {
         if (f[j] == 1) {
            printf("Block %d is already allocated.\n", j);
            allocated = 0;
            break;
         }
      }


      if (allocated) {
         for (j = st; j < st + len; j++) {
            f[j] = 1;
            printf("%d -> Allocated\n", j);
         }
         printf("The file has been allocated.\n");
```

```
        }


        printf("Do you want to enter more files? (1 for Yes / 0 for No): ");

        scanf("%d", &c);


    } while (c == 1);


    return 0;

}
```

OUTPUT:



```
Enter the starting block and length of the file: 4
2
4 -> Allocated
5 -> Allocated
The file has been allocated.
Do you want to enter more files? (1 for Yes / 0 for No): 0

--------------------------------
Process exited after 16.73 seconds with return value 0
Press any key to continue . . .
```

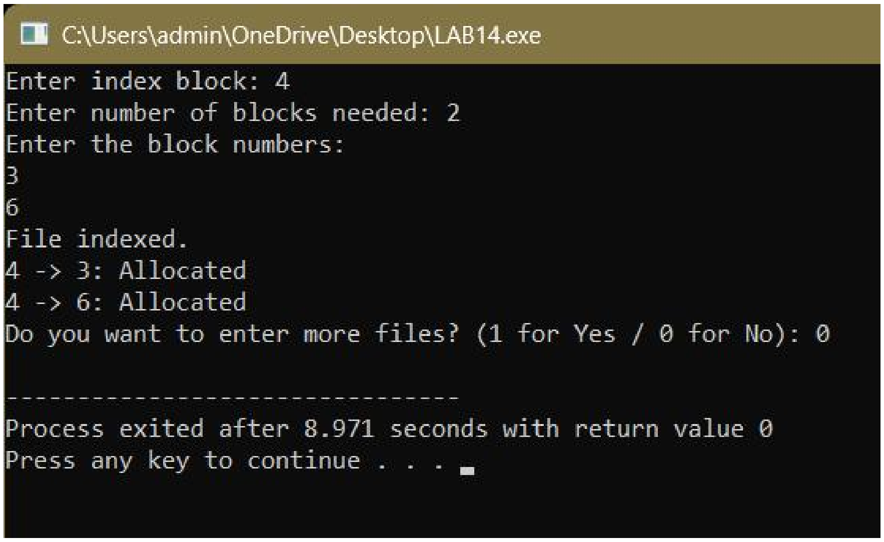
B) Indexed

```
#include <stdio.h>


int main() {

    int f[50] = {0};

    int inde[50];

    int i, n, p, c;


    do {

        printf("Enter index block: ");

        scanf("%d", &p);
```

```
if (f[p] == 1) {
    printf("Block already allocated.\n");
    continue;
}

f[p] = 1;
printf("Enter number of blocks needed: ");
scanf("%d", &n);

printf("Enter the block numbers:\n");
int valid = 1;
for (i = 0; i < n; i++) {
    scanf("%d", &inde[i]);
    if (f[inde[i]] == 1) {
        printf("Block %d already allocated.\n", inde[i]);
        valid = 0;
    }
}

if (!valid) continue;

for (i = 0; i < n; i++) {
    f[inde[i]] = 1;
}

printf("File indexed.\n");
for (i = 0; i < n; i++) {
    printf("%d -> %d: Allocated\n", p, inde[i]);
```

```
        }

        printf("Do you want to enter more files? (1 for Yes / 0 for No): ");
        scanf("%d", &c);

    } while (c == 1);

    return 0;
}
```

Output



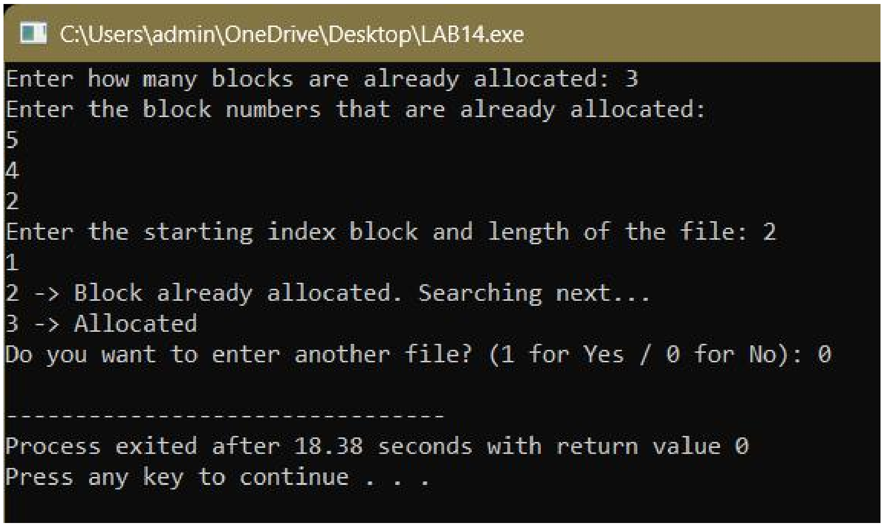
```
Enter index block: 4
Enter number of blocks needed: 2
Enter the block numbers:
3
6
File indexed.
4 -> 3: Allocated
4 -> 6: Allocated
Do you want to enter more files? (1 for Yes / 0 for No): 0

--------------------------------
Process exited after 8.971 seconds with return value 0
Press any key to continue . . .
```

C) Linked

```
#include <stdio.h>

int main() {
    int f[50] = {0};
    int i, j, p, a, st, len, c;

    printf("Enter how many blocks are already allocated: ");
    scanf("%d", &p);
    printf("Enter the block numbers that are already allocated:\n");
    for (i = 0; i < p; i++) {
        scanf("%d", &a);
        f[a] = 1;
    }

    do {
        printf("Enter the starting index block and length of the file: ");
        scanf("%d %d", &st, &len);

        int k = len;
        for (j = st; j < st + k; j++) {
            if (f[j] == 0) {
                f[j] = 1;
                printf("%d -> Allocated\n", j);
            } else {
                printf("%d -> Block already allocated. Searching next...\n", j);
                k++; // Extend search to compensate
            }
        }

        printf("Do you want to enter another file? (1 for Yes / 0 for No): ");
        scanf("%d", &c);

    } while (c == 1);

    return 0;
}
```

M.HAMZA KHAN
DT-22048

Output:



```
Enter how many blocks are already allocated: 3
Enter the block numbers that are already allocated:
5
4
2
Enter the starting index block and length of the file: 2
1
2 -> Block already allocated. Searching next...
3 -> Allocated
Do you want to enter another file? (1 for Yes / 0 for No): 0

--------------------------------
Process exited after 18.38 seconds with return value 0
Press any key to continue . . .
```