Code 15

```
#include <stdio.h>

#define MAX_BLOCKS 100

int main() {
    int disk[MAX_BLOCKS] = {0}; // 0 = free, 1 = allocated
    int start, length, i, j, files;

    printf("Enter the number of files: ");
    scanf("%d", &files);

    for(i = 0; i < files; i++) {
        printf("\nFile %d:\n", i + 1);
        printf("Enter starting block: ");
        scanf("%d", &start);

        printf("Enter length (number of blocks): ");
        scanf("%d", &length);

        // Check for out-of-bounds or overflow
        if(start < 0 || start + length > MAX_BLOCKS) {
            printf("Error: Out of disk space or invalid start block.\n");
            i--; // Retry current file
            continue;
        }

        // Check if blocks are free
        int allocated = 1;
        for(j = start; j < start + length; j++) {
            if(disk[j] == 1) {
                allocated = 0;
                break;
            }
        }

        if(allocated) {
            for(j = start; j < start + length; j++) {
                disk[j] = 1;
            }
            printf("File %d allocated from block %d to %d\n", i + 1, start, start + length - 1);
        } else {
            printf("Error: Blocks already allocated. Cannot allocate file %d.\n", i + 1);
            i--; // Retry current file
        }
    }

    // Show final allocation status
    printf("\nDisk Block Allocation:\n");
    for(i = 0; i < MAX_BLOCKS; i++) {
        printf("Block %2d: %s\n", i, disk[i] ? "Occupied" : "Free");
    }

    return 0;
}
```