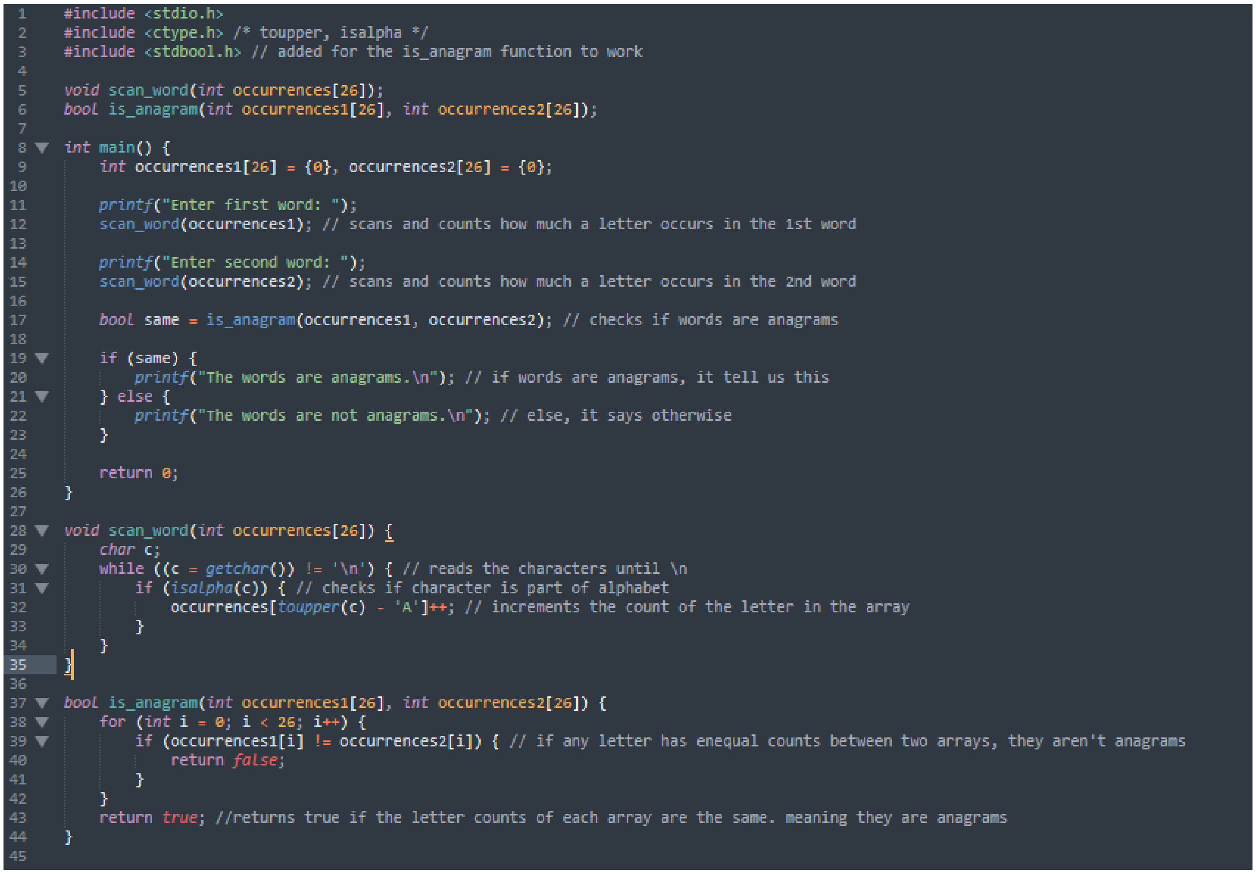
Kiano Michiko L. Soliva CMSC 21 – Section 1

202213735 23 June 2023

CMSC 21 Lec11 – Pointers and Multidimensional Arrays

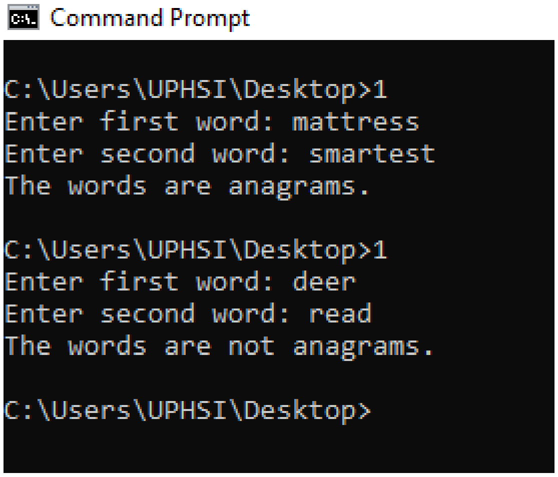
Github link:

1.



```c
#include <stdio.h>
#include <ctype.h> /* toupper, isalpha */
#include <stdbool.h> // added for the is_anagram function to work

void scan_word(int occurrences[26]);
bool is_anagram(int occurrences1[26], int occurrences2[26]);

int main() {
    int occurrences1[26] = {0}, occurrences2[26] = {0};

    printf("Enter first word: ");
    scan_word(occurrences1); // scans and counts how much a letter occurs in the 1st word

    printf("Enter second word: ");
    scan_word(occurrences2); // scans and counts how much a letter occurs in the 2nd word

    bool same = is_anagram(occurrences1, occurrences2); // checks if words are anagrams

    if (same) {
        printf("The words are anagrams.\n"); // if words are anagrams, it tell us this
    } else {
        printf("The words are not anagrams.\n"); // else, it says otherwise
    }

    return 0;
}

void scan_word(int occurrences[26]) {
    char c;
    while ((c = getchar()) != '\n') { // reads the characters until \n
        if (isalpha(c)) { // checks if character is part of alphabet
            occurrences[toupper(c) - 'A']++; // increments the count of the letter in the array
        }
    }
}

bool is_anagram(int occurrences1[26], int occurrences2[26]) {
    for (int i = 0; i < 26; i++) {
        if (occurrences1[i] != occurrences2[i]) { // if any letter has enequal counts between two arrays, they aren't anagrams
            return false;
        }
    }
    return true; //returns true if the letter counts of each array are the same. meaning they are anagrams
}

```

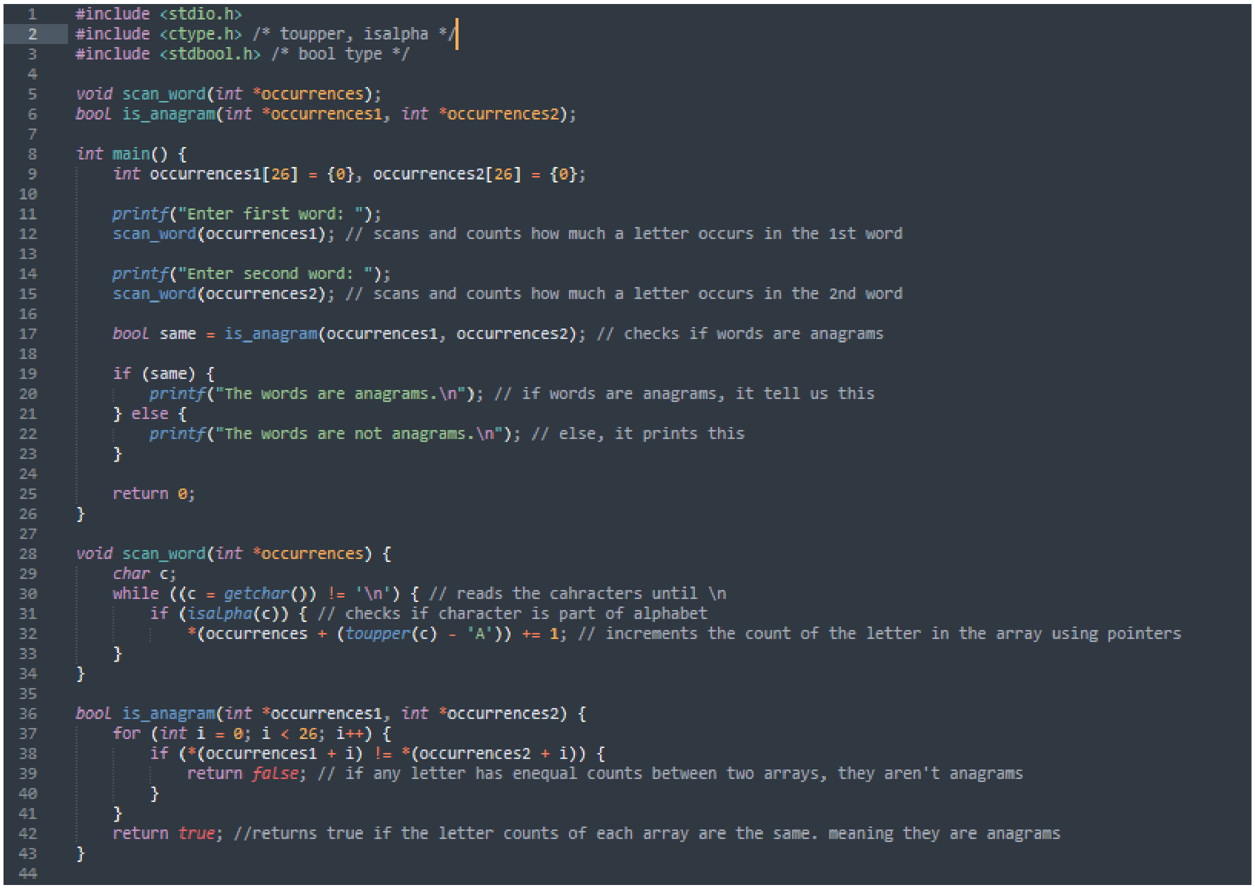
Output:



Command Prompt

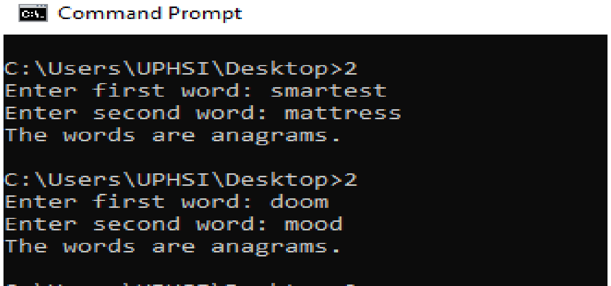
```
C:\Users\UPHSI\Desktop>1
Enter first word: mattress
Enter second word: smartest
The words are anagrams.

C:\Users\UPHSI\Desktop>1
Enter first word: deer
Enter second word: read
The words are not anagrams.

C:\Users\UPHSI\Desktop>
```

2.



```c
#include <stdio.h>
#include <ctype.h> /* toupper, isalpha */
#include <stdbool.h> /* bool type */

void scan_word(int *occurrences);
bool is_anagram(int *occurrences1, int *occurrences2);

int main() {
    int occurrences1[26] = {0}, occurrences2[26] = {0};

    printf("Enter first word: ");
    scan_word(occurrences1); // scans and counts how much a letter occurs in the 1st word

    printf("Enter second word: ");
    scan_word(occurrences2); // scans and counts how much a letter occurs in the 2nd word

    bool same = is_anagram(occurrences1, occurrences2); // checks if words are anagrams

    if (same) {
        printf("The words are anagrams.\n"); // if words are anagrams, it tell us this
    } else {
        printf("The words are not anagrams.\n"); // else, it prints this
    }

    return 0;
}

void scan_word(int *occurrences) {
    char c;
    while ((c = getchar()) != '\n') { // reads the cahracters until \n
        if (isalpha(c)) { // checks if character is part of alphabet
            *(occurrences + (toupper(c) - 'A')) += 1; // increments the count of the letter in the array using pointers
    }
}

bool is_anagram(int *occurrences1, int *occurrences2) {
    for (int i = 0; i < 26; i++) {
        if (*(occurrences1 + i) != *(occurrences2 + i)) {
            return false; // if any letter has enequal counts between two arrays, they aren't anagrams
        }
    }
    return true; //returns true if the letter counts of each array are the same. meaning they are anagrams
}
```

Output:



Command Prompt

```
C:\Users\UPHSI\Desktop>2
Enter first word: smartest
Enter second word: mattress
The words are anagrams.

C:\Users\UPHSI\Desktop>2
Enter first word: doom
Enter second word: mood
The words are anagrams.
```