



Problem Solving (C27)

هذا البرنامج التدريبي مُصاغ بعناية لتمكين المتدربين من تطوير قدراتهم الفكرية على غرار المبرمجين المحترفين، والتعاون بكفاءة ضمن فريق محترف في شركة "جمال تك" أو أي مؤسسة متعددة الجنسيات أخرى. نظرًا لأهمية اللغة الإنجليزية في بيئة العمل العالمية، يتم تقديم المحتوى التدريبي بالإنجليزية. لا يشترط إتقان اللغة بشكل كامل، لكن من الضروري امتلاك القدرة الكافية لفهم المتطلبات وتنفيذها بشكل فعال. يُمكن للمتدربين استخدام مترجم جوجل أو الاستعانة بـ "شات جي بي تي" للتغلب على أية عقبات لغوية، المهم هو الفهم الدقيق للمطلوب وتحقيقه بنجاح.

لتعظيم الاستفادة من التدريب، يُنصح بمحاولة حل التمارين بشكل مستقل لمدة ساعة واحدة على الأقل قبل الرجوع إلى الحل المرفق في نهاية الملف.

قد يتضمن الحل كودًا برمجيًا غير مفسر بعد، والغرض من ذلك هو تشجيعك على محاولة فهم الأكواد البرمجية الجديدة التي لم تتعرض لها من قبل. هذه المهارة ضرورية في سوق العمل، حيث تتطور لغات البرمجة باستمرار ويظهر كل يوم لغات جديدة. ستواجه دائمًا أكوادًا لم تدرسها من قبل، ومن المهم أن تكون قادرًا على فهمها بنفسك دون الحاجة إلى دراسة مسبقة. يمكنك الاستعانة بمحرك البحث جوجل، أو استخدام ChatGPT، أو حتى اللجوء لأصدقائك للمساعدة. الهدف الأساسي هو أن تصل إلى فهم معنى كل كود بأي طريقة ممكنة لتتمكن من إيجاد موقعك في سوق العمل.

إن وجود كود برمجي غير مفسر يشكل تحديًا يتوجب عليك إيجاد حل له. هذا النوع من التدريبات يعد جزءًا أساسيًا من تدريبات 'Problem Solving'، التي تهدف إلى تمكينك من أداء عملك بفاعلية بغض النظر عن التحديات والعقبات. هذه القدرة على حل المشكلات هي ما يتمتع به العاملون في 'جمال تك'، ومن الضروري أن تطور في نفسك هذه المهارة لتصبح عضوًا فعالًا في فريق عمل 'جمال تك'.

Gammal Tech's String Transformation System

Background:

Gammal Tech, known for its advanced software systems, often encounters challenges that require manipulating and transforming strings in complex ways. These tasks are crucial in data processing, encryption, and formatting operations. As part of their advanced programming initiatives, Gammal Tech requires a system that can take a string input and perform a series of transformations on it.

Task:

Your task is to develop a C program for Gammal Tech that performs a series of transformations on a given string. The program should read a string (up to 100 characters) and apply the following transformations:

1. Convert all lowercase letters to uppercase.
2. Replace spaces with underscores ('_').



3. For each character in the string, if it is a letter (A-Z after conversion), shift it three places forward in the alphabet (e.g., 'A' becomes 'D', 'Y' becomes 'B'). For 'Z', loop back to 'C'.

Objective:

1. Create a program that reads a string input.
2. Transform the string according to the specified rules.
3. Display the transformed string.

Constraints:

- Ensure the transformation logic is robust and handles edge cases (e.g., end of the alphabet looping).
- The program should be efficient and able to handle strings of varying lengths up to 100 characters.

Sample Input:

```
Enter a string: Hello World 2023
```

Sample Output:

```
Transformed string: KHOOR_ZRUOG_2023
```

لتعزيز الاستفادة من التدريب، يُنصح بمحاولة حل التمرين بشكل مستقل لمدة ساعة واحدة على الأقل قبل الرجوع إلى الحل المرفق



C Programming Solution:

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

void transformString(char *str) {
    for (int i = 0; i < strlen(str); i++) {
        // Convert to uppercase
        str[i] = toupper(str[i]);

        // Replace spaces with underscores
        if (str[i] == ' ') {
            str[i] = '_';
        }

        // Shift letters three places forward
        if (str[i] >= 'A' && str[i] <= 'Z') {
            if (str[i] > 'W') {
                str[i] = 'A' + (str[i] - 'W' - 1);
            } else {
                str[i] += 3;
            }
        }
    }
}

int main() {
    char text[100];

    // Reading the string input
    printf("Enter a string: ");
    fgets(text, sizeof(text), stdin);
    text[strlen(text), "\n"] = 0; // Remove newline character

    // Transforming the string
    transformString(text);

    // Displaying the transformed string
    printf("Transformed string: %s\n", text);

    return 0;
}
```