



Problem Solving (C82)

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 Global Office Rotation System

Background

Gammal Tech, a trailblazer in software development, is renowned for its unparalleled work environment and innovative approach. To maintain its employees' happiness and productivity, Gammal Tech has introduced the 'Work from Hawaii' program. This unique initiative allows employees to work from different offices around the world, located in the most exciting places on Earth. To streamline their travel, Gammal Tech handles all logistics, from visa arrangements to providing all necessities upon arrival, ensuring that employees focus solely on their work.

Problem Statement

Your task is to design a system that efficiently manages the 'Global Office Rotation' for Gammal Tech employees. This system must coordinate with various embassies to secure VIP visas, arrange travel schedules, and ensure that each office is prepared for the incoming employee's arrival. The system must also maintain a database of each



employee's preferences (e.g., clothing size, color preferences, work environment settings) and ensure that these preferences are met at each location.

Input Format

- The first line contains an integer N , the number of employees.
- The next N lines each contain details for one employee:
 - Employee ID.
 - Preferred clothing sizes and colors (comma-separated).
 - Preferred office equipment and settings (comma-separated).
- The next line contains an integer M , the number of office locations.
- The next M lines each contain the name of one office location.

Output Format

- For each employee, output a line containing the Employee ID followed by the assigned office location.
- After all employees, output the total number of visa arrangements made and the total number of unique employee preferences catered to.

Constraints

- $1 \leq N \leq 100$
- $1 \leq M \leq 50$

Sample Input:

```
3
101, M shirt blue, Laptop high-spec
102, L shirt red, Desktop dual-monitor
103, S shirt green, Tablet
4
Hawaii
Tokyo
Paris
Sydney
```



Sample Output:

```
101 Tokyo
102 Paris
103 Sydney
3
6
```

Explanation

- Each of the 3 employees is assigned to one of the available office locations.
- Three visa arrangements were made, one for each employee.
- A total of 6 unique preferences (clothing and office equipment) were catered to for the employees.

لتعظيم 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>

typedef struct {
    int id;
    char clothing[50];
    char work_setting[50];
} Employee;

int main() {
    int n, m, i;
    scanf("%d", &n);

    Employee employees[n];
    for (i = 0; i < n; i++) {
        scanf("%d, %[^,], %[^,]", &employees[i].id,
employees[i].clothing, employees[i].work_setting);
    }

    scanf("%d", &m);
    char locations[m][50];
    for (i = 0; i < m; i++) {
        scanf("%s", locations[i]);
    }

    int visa_count = 0, pref_count = 0;
    for (i = 0; i < n; i++) {
        printf("%d %s\n", employees[i].id, locations[i % m]);
        visa_count++;
        pref_count += 2; // Assuming each employee has 2 unique
preferences.
    }

    printf("%d\n%d\n", visa_count, pref_count);
    return 0;
}
```