

TEST 3. STRING OPERATIONS

1. **Loại bỏ ký tự trùng nhau (Microsoft).** Cho chuỗi ký tự `str[]` có độ dài n ($n \leq 10000$). Hãy loại bỏ các ký tự trùng lặp trong chuỗi.

Ví dụ:

Input:	Output:
ABCABC	ABC
GEEKSFORGEEKS	GEKSFOR

2. **Pangram String.** Một chuỗi ký tự được gọi là một Pangram nếu nó chứa đầy đủ các ký tự 'a',...,z hoặc 'A',..., 'Z' hoặc cả hai. Cho chuỗi ký tự `str[]`, hãy kiểm tra xem `str[]` có phải là một Pangram hay không?

Ví dụ:

Input:	Output:
ABCDEF	NO
ABCDEFGHIJKLMNOPQRSTUVWXYZ	YES

3. Cho chuỗi ký tự `str[]`. Hãy tìm ký tự xuất hiện nhiều lần nhất và số lần xuất hiện ký tự đó trong `str[]`. Trong trường hợp có nhiều ký tự xuất hiện nhiều lần nhất, hãy đưa ra ký tự đầu tiên.

Ví dụ:

Input:	Output:
ABCDCCC	C 4
ABC ABC ABC	A 3

4. Cho chuỗi ký tự `str[]`. Hãy tìm tập ký tự và số lần xuất hiện các ký tự trong chuỗi.

Ví dụ:

Input:	Output:
ABCABC	A 2
	B 2
	C 2

5. Cho chuỗi ký tự `str1[]` và chuỗi `str2[]`. Hãy tìm tập ký tự và số lần xuất hiện mỗi ký tự trong cả hai chuỗi?

Ví dụ:

Input:	Output:
ABCDEFABCDEF	A 4
ABCGHABCGH	B 4
	C 4
	D 2
	E 2
	F 2
	G 2
	H 2

6. Cho chuỗi ký tự `str1[]` và chuỗi `str2[]`. Hãy tìm tập ký tự và số lần xuất hiện mỗi ký tự trong `str1[]` và `str2[]`.

Ví dụ:

Input:	Output:
ABCDEFABCDEF	A 4
ABCGHABCGH	B 4
	C 4

7. Cho xâu ký tự str1[] và xâu str2[]. Hãy tìm tập ký tự và số lần xuất hiện ký tự trong str1[] nhưng không xuất hiện trong xâu str2[].

Ví dụ:

Input:	Output:
ABCDEFABCDEF	D 2
ABCGHABCGH	E 2
	F 2

8. Ta gọi một từ là dãy các ký tự không chứa ký tự rỗng ' ', tab '\t', xuống dòng '\n', về đầu dòng '\r', và kết thúc dòng '\0'. Cho xâu ký tự str[], hãy tìm tập từ và số lần xuất hiện mỗi từ.

Ví dụ:

Input:	Output:
AB AC AD AB AC AD	3
	AB 2
	AC 2
	AD 2

9. Ta gọi một từ là dãy các ký tự không chứa ký tự rỗng ' ', tab '\t', xuống dòng '\n', về đầu dòng '\r', và kết thúc dòng '\0'. Cho xâu ký tự str1[], và xâu str2[], hãy tìm tập từ và số lần xuất hiện mỗi từ trong cả hai xâu.

Ví dụ:

Input:	Output:
AB AC AD AB AC AD	6
AB AC AD AG AH AK	AB 3
	AC 3
	AD 3
	AG 1
	AH 1
	AK 1

10. Ta gọi một từ là dãy các ký tự không chứa ký tự rỗng ' ', tab '\t', xuống dòng '\n', về đầu dòng '\r', và kết thúc dòng '\0'. Cho xâu ký tự str1[], và xâu str2[], hãy tìm tập từ và số lần xuất hiện mỗi từ trong str1[] và str2[].

Ví dụ:

Input:	Output:
AB AC AD AB AC AD	3
AB AC AD AG AH AK	AB 3
	AC 3
	AD 3

11. Ta gọi một từ là dãy các ký tự không chứa ký tự rỗng ' ', tab '\t', xuống dòng '\n', về đầu dòng '\r', và kết thúc dòng '\0'. Cho xâu ký tự str1[], và xâu str2[], hãy tìm tập từ và số lần xuất hiện mỗi từ trong str1[] nhưng không xuất hiện trong str2[].

Ví dụ:

Input:	Output:
AB AC AD AE AF AG	2
AB AC AD AG AH AK	AE 1
	AF 1

12. **Anagram Strings (Microsoft).** Hai xâu ký tự s1[], s2[] được gọi là Anagram của nhau nếu ta có thể sắp đặt lại các ký tự trong xâu s1[] để nhận được xâu s2[]. Ví dụ xâu s1[]="ABCD" là Anagram của xâu s2[]="BCDA". Cho hai xâu ký tự s1[] và s2[], hãy kiểm tra xem s1[] có phải là Anagram của s2[] hay không.

Ví dụ:

Input:	Output:
ABCD	YES
DCBA	

13. **Anagram Numbers** (Samsung). Hai số a và b được gọi là Anagram của nhau nếu ta có thể sắp đặt lại các bit 0, 1 trong biểu diễn nhị phân của số a để trở thành biểu diễn nhị phân của số b. Ví dụ a = 8 = 1000, b = 4 = 0100 là Anagram của nhau. Cho cặp số nguyên dương a và b, hãy kiểm tra xem chúng có phải là Anagram của nhau hay không?

Ví dụ:

Input:	Output:
8 4	YES
12 5	YES
12 7	NO

14. **Anagram Palindrom Strings** (Facebook). Một chuỗi ký tự được gọi là Palindrom nếu nó là một chuỗi đối xứng. Ví dụ chuỗi s[] = "ABCBA" là một Palindrom. Cho chuỗi ký tự s[], hãy cho biết ta có thể tạo nên một Palindrom bằng cách đảo ngược các ký tự ở những vị trí khác nhau trong s[] hay không?

Ví dụ:

Input:	Output:
ABCDEFGHIJABCDEFGHIJ	YES
ABCDEFABCDEFABCDEF	NO

15. **K-grams String**. Hai chuỗi ký tự được gọi là một k-grams nếu hai chuỗi thỏa mãn điều kiện:

- Hai chuỗi có cùng tập các ký tự.
- Hai chuỗi có thể trở thành Anagrams bằng cách thay đổi ít nhất k ký tự trong một chuỗi.

Cho hai chuỗi ký tự s1[] và s2[], hãy cho biết hai chuỗi có phải là k-grams hay không?

Ví dụ:

Input:	Output:
3 (k=3)	YES
Anagram (chuỗi s1[])	NO
Grammar (chuỗi s2[])	
1 (k=3)	
ABCD (chuỗi s1[])	
AGHK (chuỗi s2[])	

16. **Palindrom Substring Queries** (Samsung). Cho chuỗi ký tự S[] có độ dài n và một số câu hỏi Q = {q₁, q₂, ..., q_k}. Mỗi câu hỏi q_i là một bộ đôi s, t (0 ≤ s ≤ t ≤ n). Hãy cho biết chuỗi con tương ứng với mỗi câu hỏi q_i ∈ Q có tạo nên một Palindrom hay không?

Ví dụ:

Input:	Output:
abaaabaaaba	YES
0 10	NO
5 8	NO
2 5	YES
5 9	

17.