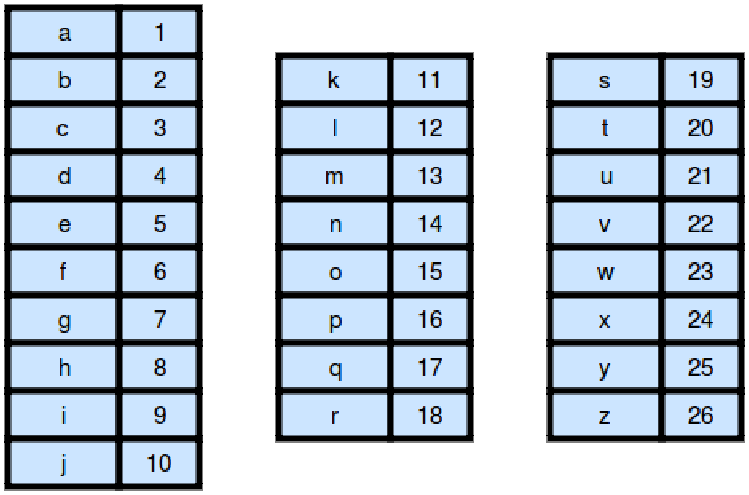
# Weighted Uniform Strings

A weighted string is a string of lowercase English letters where each letter has a *weight*. Character weights are $1$ to $26$ from $a$ to $z$ as shown below:

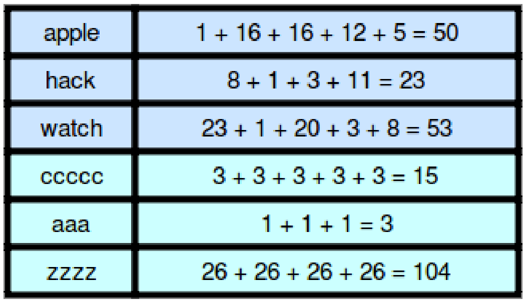


| | |
|---|---|
| a | 1 |
| b | 2 |
| c | 3 |
| d | 4 |
| e | 5 |
| f | 6 |
| g | 7 |
| h | 8 |
| i | 9 |
| j | 10 |

| | |
|---|---|
| k | 11 |
| l | 12 |
| m | 13 |
| n | 14 |
| o | 15 |
| p | 16 |
| q | 17 |
| r | 18 |

| | |
|---|---|
| s | 19 |
| t | 20 |
| u | 21 |
| v | 22 |
| w | 23 |
| x | 24 |
| y | 25 |
| z | 26 |

- The *weight of a string* is the sum of the weights of its characters. For example:



| | |
|---|---|
| apple | 1 + 16 + 16 + 12 + 5 = 50 |
| hack | 8 + 1 + 3 + 11 = 23 |
| watch | 23 + 1 + 20 + 3 + 8 = 53 |
| ccccc | 3 + 3 + 3 + 3 + 3 = 15 |
| aaa | 1 + 1 + 1 = 3 |
| zzzz | 26 + 26 + 26 + 26 = 104 |

- A *uniform string* consists of a single character repeated zero or more times. For example, `ccc` and `a` are uniform strings, but `bcb` and `cd` are not.

Given a string, $s$, let $U$ be the set of weights for all possible uniform contiguous substrings of string $s$. There will be $n$ queries to answer where each query consists of a single integer. Create a return array where for each query, the value is `Yes` if $query[i] \in U$. Otherwise, append `No`.

**Note:** The $\in$ symbol denotes that $x[i]$ is an element of set $U$.

**Example**

$s =$ 'abbcccdddd'
$queries = [1, 7, 5, 4, 15]$.

Working from left to right, weights that exist are:

```
string  weight
a       1
b       2
bb      4
c       3
```

```
cc      6
ccc     9
d       4
dd      8
ddd     12
dddd    16
```

Now for each value in $queries$, see if it exists in the possible string weights. The return array is `['Yes', 'No', 'No', 'Yes', 'No']`.

### Function Description

Complete the *weightedUniformStrings* function in the editor below.

weightedUniformStrings has the following parameter(s):
- *string s:* a string
- *int queries[n]:* an array of integers

### Returns

- *string[n]:* an array of strings that answer the queries

### Input Format

The first line contains a string $s$, the original string.
The second line contains an integer $n$, the number of queries.
Each of the next $n$ lines contains an integer $queries[i]$, the weight of a uniform subtring of $s$ that may or may not exist.

### Constraints

- $1 \leq lengthofs, n \leq 10^5$
- $1 \leq queries[i] \leq 10^7$
- $s$ will only contain lowercase English letters, ascii[a-z].

### Sample Input 0

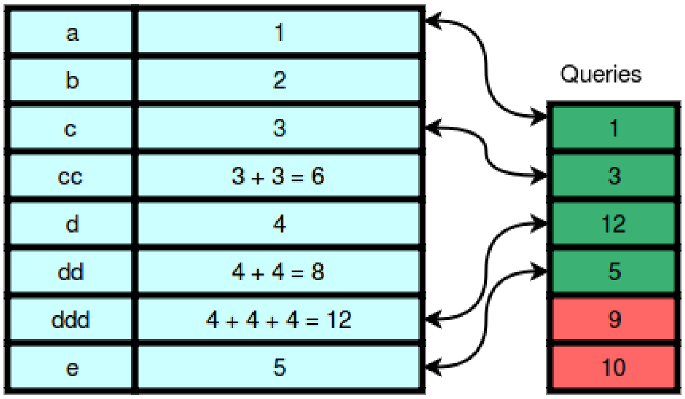
```
abccddde
6
1
3
12
5
9
10
```

### Sample Output 0

```
Yes
Yes
Yes
Yes
No
No
```

### Explanation 0

The weights of every possible *uniform substring* in the string `abccddde` are shown below:



| | |
|---|---|
| a | 1 |
| b | 2 |
| c | 3 |
| cc | 3 + 3 = 6 |
| d | 4 |
| dd | 4 + 4 = 8 |
| ddd | 4 + 4 + 4 = 12 |
| e | 5 |

We print `Yes` on the first four lines because the first four queries match weights of uniform substrings of $s$. We print `No` for the last two queries because there are no uniform substrings in $s$ that have those weights.

Note that while `de` is a substring of $s$ that would have a weight of $9$, it is *not a uniform substring*.

Note that we are only dealing with contiguous substrings. So `ccc` is not a substring of the string `ccxxc`.

### Sample Input 1

```
aaabbbbcccddd
5
9
7
8
12
5
```

### Sample Output 1

```
Yes
No
Yes
Yes
No
```