30. Substring with Concatenation of All Words

题目描述：https://leetcode.com/problems/substring-with-concatenation-of-all-words/

> 给定一个长单词，给定一个字符串数组，求数组中combine起来的单词在长单词的位置。
> 例如：
>
> ```
> s: "barfoothefoobarman"
> words: ["foo", "bar"]
> You should return the indices: [0,9].
> ```

解题思路：

用unordered_map存储着每个单词该出现的次数。 从长单词的每个位置开始往后看n*l长度，把每个单词看到的次数存储下来。和第一个对比。

代码：

```
class Solution {
public:
    vector<int> findSubstring(string s, vector<string>& words) {
        int n = words.size(), len = s.size();
        vector<int> res;
        if(n == 0) return {};
        unordered_map<string, int> expect;
        for(string item: words) {
            expect[item]++;
        }
        int l = words[0].size();
        for(int i = 0; i+n*l-1 < len; i++) {
            unordered_map<string, int> seen;
            int j = 0;
            for(; j < n; j++) {
                string t = s.substr(i+j*l, l);
                if(expect.find(t) == expect.end()) {
                    break;
                }
                seen[t] += 1;
                if(seen[t] > expect[t]) {
                    break;
                }
            }
            if(j == n) {
                res.push_back(i);
            }
        }
        return res;
    }
};
```