

Follow me for more interesting programming questions and RAW Code visit my GitHub profile:
<https://www.github.com/shubhamthrills> <https://www.linkedin.com/in/shubhamsagar>

Subscribe our YouTube Channel for more videos: <https://www.youtube.com/channel/UCjLMu9mayAibQST1eWwq0cg>

Leetcode May Challenge DAY: 17

1. Python

class Solution(object):

def findAnagrams(self, s, p):

if len(s) < len(p):

return []

li1 = [0]*26

li2 = [0]*26

res = []

for i in range(len(p)):

li1[ord(p[i])-ord('a')] += 1

li2[ord(s[i])-ord('a')] += 1

if li1 == li2:

res.append(0)

for i in range(1, len(s)-len(p)+1):

li2[ord(s[i-1])-ord('a')] -= 1

li2[ord(s[i+len(p)-1])-ord('a')] += 1

if li1 == li2:

res.append(i)

return res

Follow me for more interesting programming questions and RAW Code visit my GitHub profile:
<https://www.github.com/shubhamthrills> <https://www.linkedin.com/in/shubhamsagar>

Subscribe our YouTube Channel for more videos: <https://www.youtube.com/channel/UCjLMu9mayAibQST1eWwq0cg>

2. C++

class Solution {

public:

```
vector<int> findAnagrams(string s, string p)
{
    vector<int> result;

    if (s.size() < p.size())
        return result;

    int count[ 26 ] = { 0 };
    for( int i = 0; i < p.size() - 1; i++ )
        count[ (int)s[ i ] - 'a' ]++;

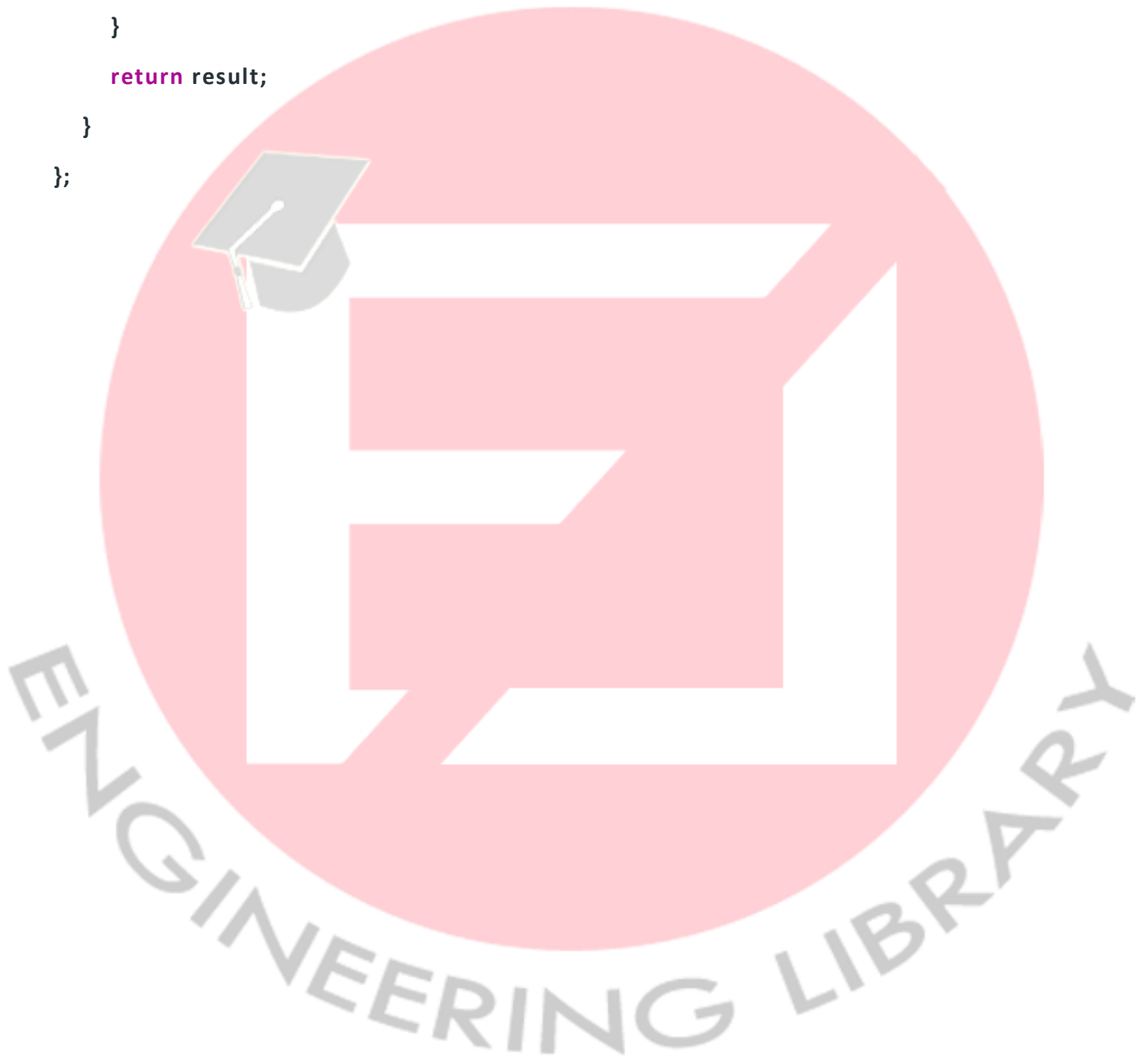
    int patCount[ 26 ] = { 0 };
    for( int k = 0; k < p.size(); k++ )
        patCount[ p[ k ] - 'a' ]++;

    for( int i = 0, j = p.size()-1; i + p.size() <= s.size(); i++, j++ )
    {
        count[ s[ j ] - 'a' ]++;
        bool flag = true;
        for( int k = 0; k < 26; k++ )
        {
            if( count[ k ] != patCount[ k ] )
            {
                flag = false;
                break;
            }
        }
    }
}
```

Follow me for more interesting programming questions and RAW Code visit my GitHub profile:
<https://www.github.com/shubhamthrills> <https://www.linkedin.com/in/shubhamsagar>

Subscribe our YouTube Channel for more videos: <https://www.youtube.com/channel/UCJLMu9mayAibQST1eWwq0cg>

```
    }  
  }  
  if( flag ) result.push_back( i );  
  count[ s[ i ] - 'a' ]--;  
}  
return result;  
}  
};
```



3. JAVA

```
public List<Integer> findAnagrams(String s, String p) {  
    List<Integer> anagrams = new ArrayList<>();  
    if (p == null || s == null || s.length() == 0 || p.length() == 0)  
        return anagrams;  
  
    int[] target = new int[26];  
    for (int i = 0; i < p.length(); i++) {  
        target[p.charAt(i) - 'a']++;  
    }  
  
    StringBuilder sb = new StringBuilder();  
    int hops = p.length();  
  
    for (int i = 0; i < s.length() - hops + 1; i++) {  
        sb.setLength(0);  
        sb.append(s, i, i+hops);  
        if (matches(sb, target)){  
            anagrams.add(i);  
        }  
    }  
  
    return anagrams;  
}
```

```
private boolean matches(StringBuilder sb, int[] target){
```

```
    int[] value = new int[26];
```

Follow me for more interesting programming questions and RAW Code visit my GitHub profile:
<https://www.github.com/shubhamthrills> <https://www.linkedin.com/in/shubhamsagar>

Subscribe our YouTube Channel for more videos: <https://www.youtube.com/channel/UCjLMu9mayAibQST1eWwq0cg>

```
for (int i = 0; i < sb.length(); i++) {  
    value[sb.charAt(i) - 'a']++;  
}  
  
for (int i = 0; i < value.length; i++) {  
    if (value[i] != target[i]) return false;  
}  
  
return true;  
}
```



ENGINEERING LIBRARY