

Day-16:String-2

Problem Statement: Z-Function

```
def calculate_z_function(s):
```

```
    n = len(s)
```

```
    z = [0] * n
```

```
    left = right = 0
```

```
    for i in range(1, n):
```

```
        if i <= right:
```

```
            k = i - left
```

```
            if z[k] < right - i + 1:
```

```
                z[i] = z[k]
```

```
            else:
```

```
                left = i
```

```
                while right < n and s[right] == s[right - left]:
```

```
                    right += 1
```

```
                z[i] = right - left
```

```
                right -= 1
```

```
    else:
```

```
        left = right = i
```

```
        while right < n and s[right] == s[right - left]:
```

```
            right += 1
```

```
        z[i] = right - left
```

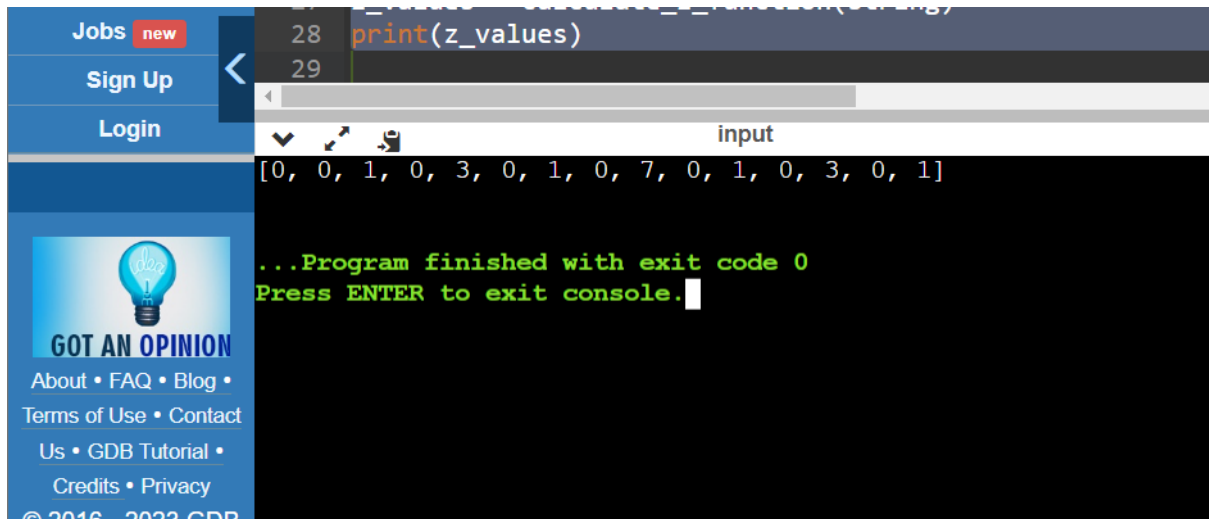
```
        right -= 1
```

```
return z
```

```
string = "abacabadabacaba"
```

```
z_values = calculate_z_function(string)
```

```
print(z_values)
```



The screenshot shows a web browser interface. On the left is a blue sidebar with navigation links: "Jobs new", "Sign Up", "Login", "GOT AN OPINION" (with a lightbulb icon), "About • FAQ • Blog •", "Terms of Use • Contact", "Us • GDB Tutorial •", "Credits • Privacy", and "© 2016 - 2023 GDB". The main content area displays a terminal window. The terminal shows the execution of a program that prints the Z-function values for the string "abacabadabacaba". The output is: [0, 0, 1, 0, 3, 0, 1, 0, 7, 0, 1, 0, 3, 0, 1]. Below the output, the terminal shows "...Program finished with exit code 0" and "Press ENTER to exit console.".

```
28 print(z_values)
29
```

input

```
[0, 0, 1, 0, 3, 0, 1, 0, 7, 0, 1, 0, 3, 0, 1]
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

Problem Statement: KMP algo/LPS(pi) array

```
def compute_lps(pattern):
```

```
    length = len(pattern)
```

```
    lps = [0] * length
```

```
    j = 0
```

```
    for i in range(1, length):
```

```
        while j > 0 and pattern[i] != pattern[j]:
```

```
            j = lps[j-1]
```

```
        if pattern[i] == pattern[j]:
```

```
            j += 1
```

```
        lps[i] = j
```

```
    return lps
```

```
def kmp_search(text, pattern):
```

```
    n = len(text)
```

```
    m = len(pattern)
```

```
    lps = compute_lps(pattern)
```

```
    i = j = 0
```

```
    indices = []
```

```
    while i < n:
```

```
        if pattern[j] == text[i]:
```

```
            i += 1
```

```
            j += 1
```

```
        if j == m:
```

```
            indices.append(i - j)
```

```
            j = lps[j-1]
```

```
        else:
```

```
            if j != 0:
```

```
                j = lps[j-1]
```

```
            else:
```

```
                i += 1
```

```
    return indices
```

```
text = "ABABDABACDABABCABAB"
```

```
pattern = "ABABCABAB"
```

```
lps_array = compute_lps(pattern)
```

```
print("LPS Array:", lps_array)
```

```

matches = kmp_search(text, pattern)

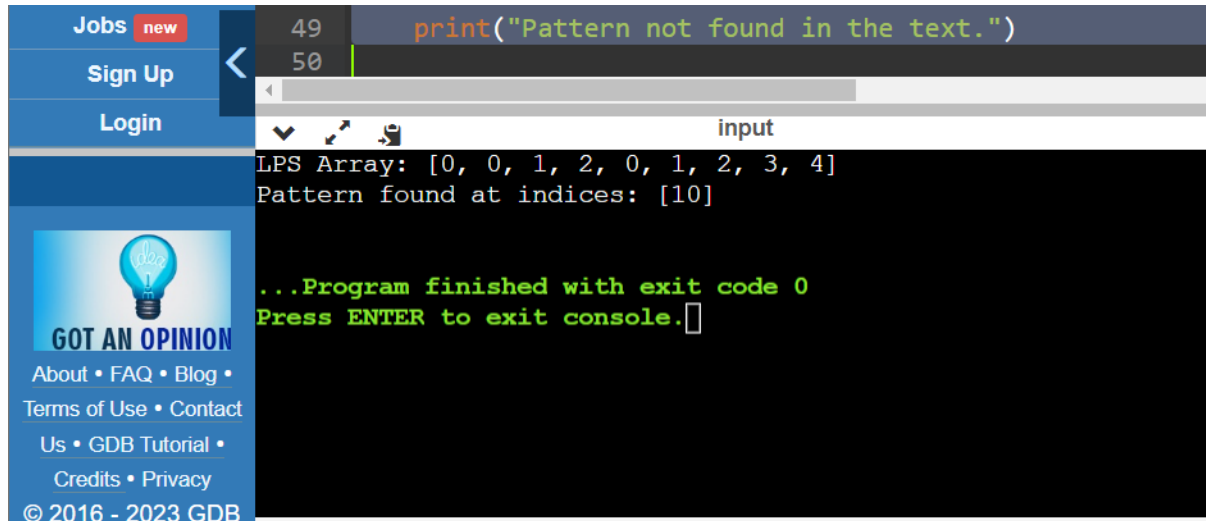
if matches:

    print("Pattern found at indices:", matches)

else:

    print("Pattern not found in the text.")

```



The screenshot shows a web browser interface. On the left is a blue sidebar with navigation links: 'Jobs new', 'Sign Up', 'Login', and a section titled 'GOT AN OPINION' with links for 'About', 'FAQ', 'Blog', 'Terms of Use', 'Contact Us', 'GDB Tutorial', 'Credits', and 'Privacy'. The main content area has a dark background and displays the following text: 'LPS Array: [0, 0, 1, 2, 0, 1, 2, 3, 4]', 'Pattern found at indices: [10]', and '...Program finished with exit code 0'. At the bottom, it says 'Press ENTER to exit console.' with a cursor icon.

Problem Statement: Minimum Characters required to make a String Palindromic

```

def min_chars_to_palindrome(string):

    if string == string[::-1]:

        return 0

    n = len(string)

    for i in range(n, 0, -1):

        if string[:i] == string[i - 1::-1]:

            return n - i

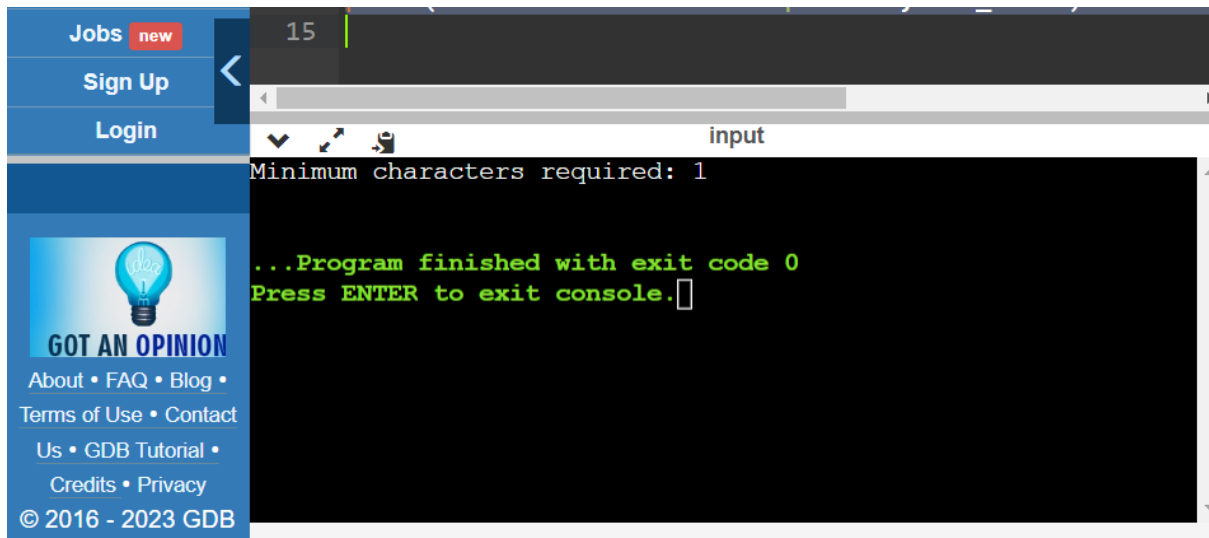
    return n - 1

input_string = "ab"

min_chars = min_chars_to_palindrome(input_string)

print("Minimum characters required:", min_chars)

```



Problem Statement: Check for Anagrams

```
def are_anagrams(str1, str2):
```

```
    str1 = str1.lower().replace(" ", "")
```

```
    str2 = str2.lower().replace(" ", "")
```

```
    if len(str1) != len(str2):
```

```
        return False
```

```
    char_count1 = {}
```

```
    char_count2 = {}
```

```
    for char in str1:
```

```
        char_count1[char] = char_count1.get(char, 0) + 1
```

```
for char in str2:
```

```
    char_count2[char] = char_count2.get(char, 0) + 1
```

```
if char_count1 == char_count2:
```

```
    return True
```

```
else:
```

```
    return False
```

```
string1 = "listen"
```

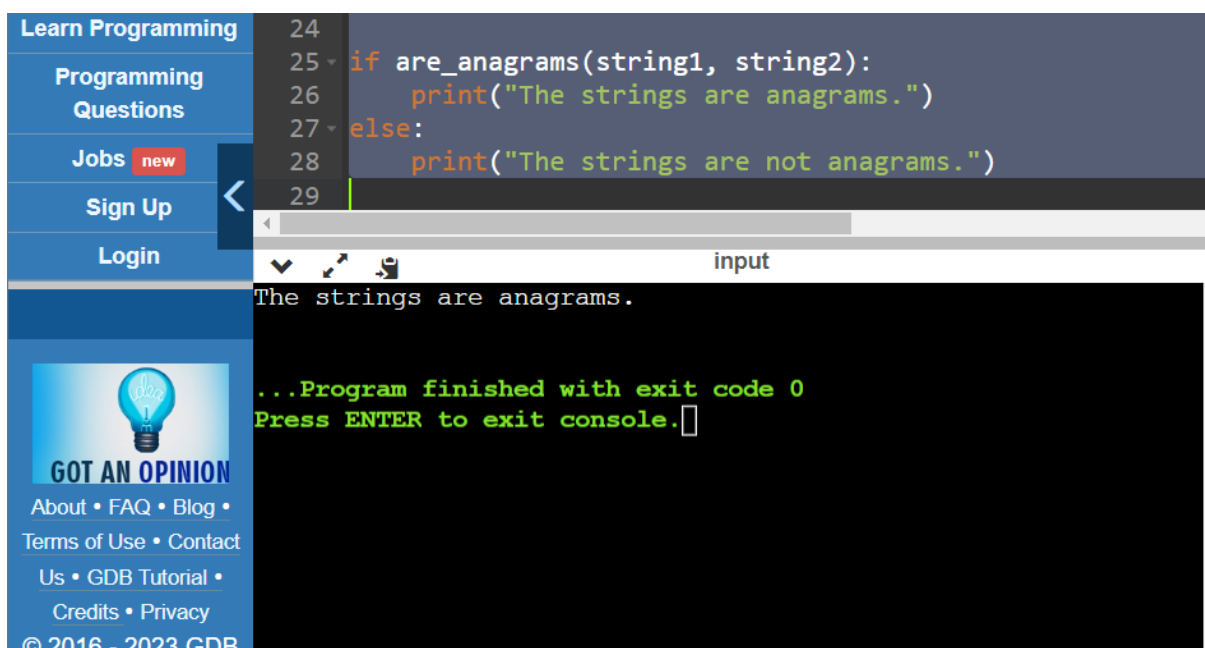
```
string2 = "silent"
```

```
if are_anagrams(string1, string2):
```

```
    print("The strings are anagrams.")
```

```
else:
```

```
    print("The strings are not anagrams.")
```



The screenshot shows a web application interface. On the left is a blue sidebar with navigation links: "Learn Programming", "Programming Questions", "Jobs" (with a red "new" badge), "Sign Up", and "Login". Below these is a section titled "GOT AN OPINION" with a lightbulb icon and links for "About", "FAQ", "Blog", "Terms of Use", "Contact Us", "GDB Tutorial", "Credits", and "Privacy". At the bottom of the sidebar is the copyright notice "© 2016 - 2023 GDB". The main area on the right contains a code editor with Python code for an anagram checker. The code is as follows:

```
24
25 if are_anagrams(string1, string2):
26     print("The strings are anagrams.")
27 else:
28     print("The strings are not anagrams.")
29
```

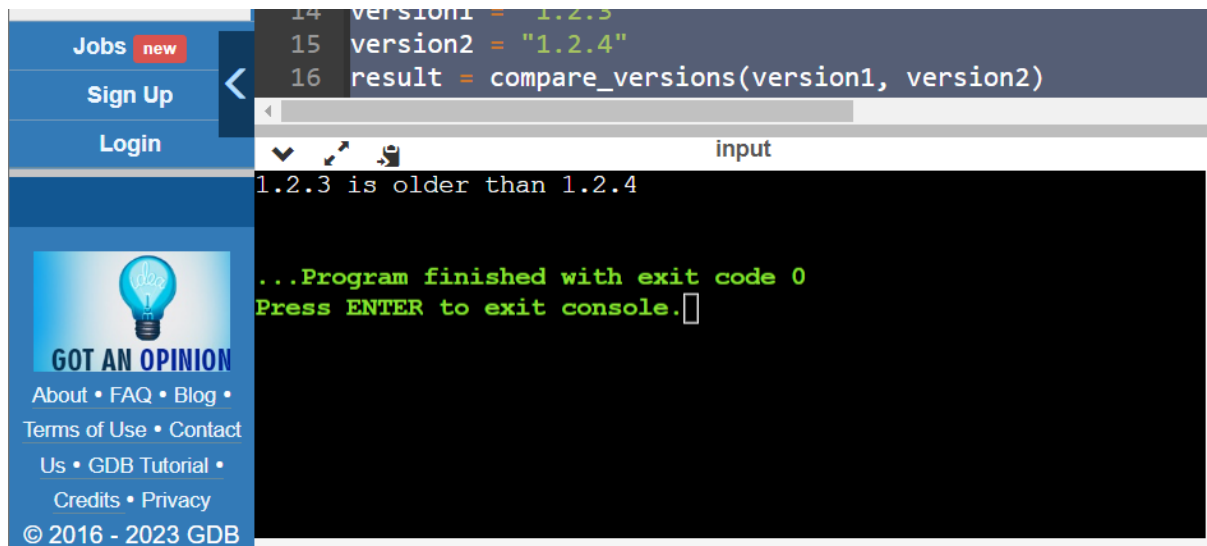
Below the code editor is a terminal window. The terminal shows the output "The strings are anagrams." and a message "...Program finished with exit code 0" followed by "Press ENTER to exit console." with a cursor.

Problem Statement: Compare version numbers

```
def compare_versions(version1, version2):  
    v1 = version1.split('.')  
    v2 = version2.split('.')  
    for i in range(max(len(v1), len(v2))):  
        num1 = int(v1[i]) if i < len(v1) else 0  
        num2 = int(v2[i]) if i < len(v2) else 0  
  
        if num1 < num2:  
            return -1  
        elif num1 > num2:  
            return 1  
    return 0  
  
version1 = "1.2.3"  
version2 = "1.2.4"  
result = compare_versions(version1, version2)  
  
if result < 0:  
    print(f"{version1} is older than {version2}")  
elif result > 0:  
    print(f"{version1} is newer than {version2}")
```

else:

```
print(f"{version1} and {version2} are the same version")
```



The screenshot shows a web browser interface. On the left is a blue sidebar with the following links: 'Jobs' (with a 'new' badge), 'Sign Up', 'Login', 'GOT AN OPINION' (with a lightbulb icon), and a list of links: 'About', 'FAQ', 'Blog', 'Terms of Use', 'Contact Us', 'GDB Tutorial', 'Credits', and 'Privacy'. At the bottom of the sidebar is '© 2016 - 2023 GDB'. The main content area has a dark background. It shows code lines 14, 15, and 16:

```
14 version1 = "1.2.3"
15 version2 = "1.2.4"
16 result = compare_versions(version1, version2)
```

 Below the code is a terminal window titled 'input' showing the output:

```
1.2.3 is older than 1.2.4
...Program finished with exit code 0
Press ENTER to exit console.
```

Problem Statement : Count and say

```
def countAndSay(n):
```

```
    if n == 1:
```

```
        return "1"
```

```
    prev_seq = countAndSay(n - 1)
```

```
    count = 1
```

```
    result = ""
```

```
    for i in range(len(prev_seq)):
```

```
        if i + 1 < len(prev_seq) and prev_seq[i] == prev_seq[i + 1]:
```

```
            count += 1
```

```
        else:
```

```
            result += str(count) + prev_seq[i]
```



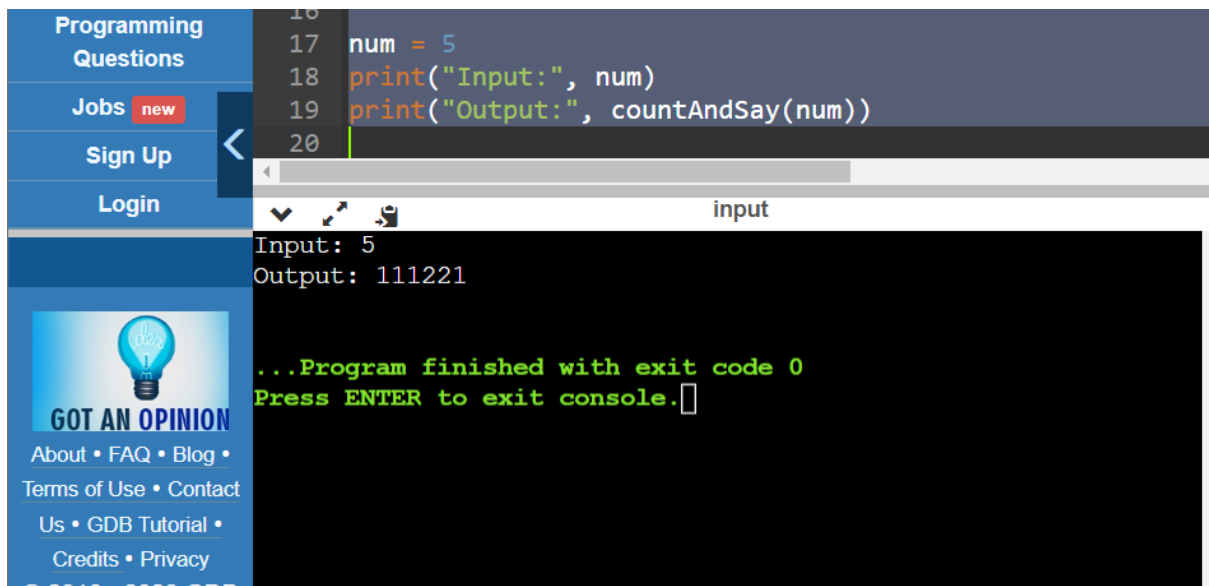
```
count = 1
```

```
return result
```

```
num = 5
```

```
print("Input:", num)
```

```
print("Output:", countAndSay(num))
```



The screenshot shows a web application interface. On the left is a blue sidebar with navigation links: "Programming Questions", "Jobs new", "Sign Up", and "Login". Below these is a section titled "GOT AN OPINION" with a lightbulb icon and links for "About", "FAQ", "Blog", "Terms of Use", "Contact Us", "GDB Tutorial", "Credits", and "Privacy". The main area on the right contains a code editor with Python code:

```
16  
17 num = 5  
18 print("Input:", num)  
19 print("Output:", countAndSay(num))  
20
```

 Below the code editor is a terminal window titled "input" showing the program's execution:

```
Input: 5  
Output: 111221  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```