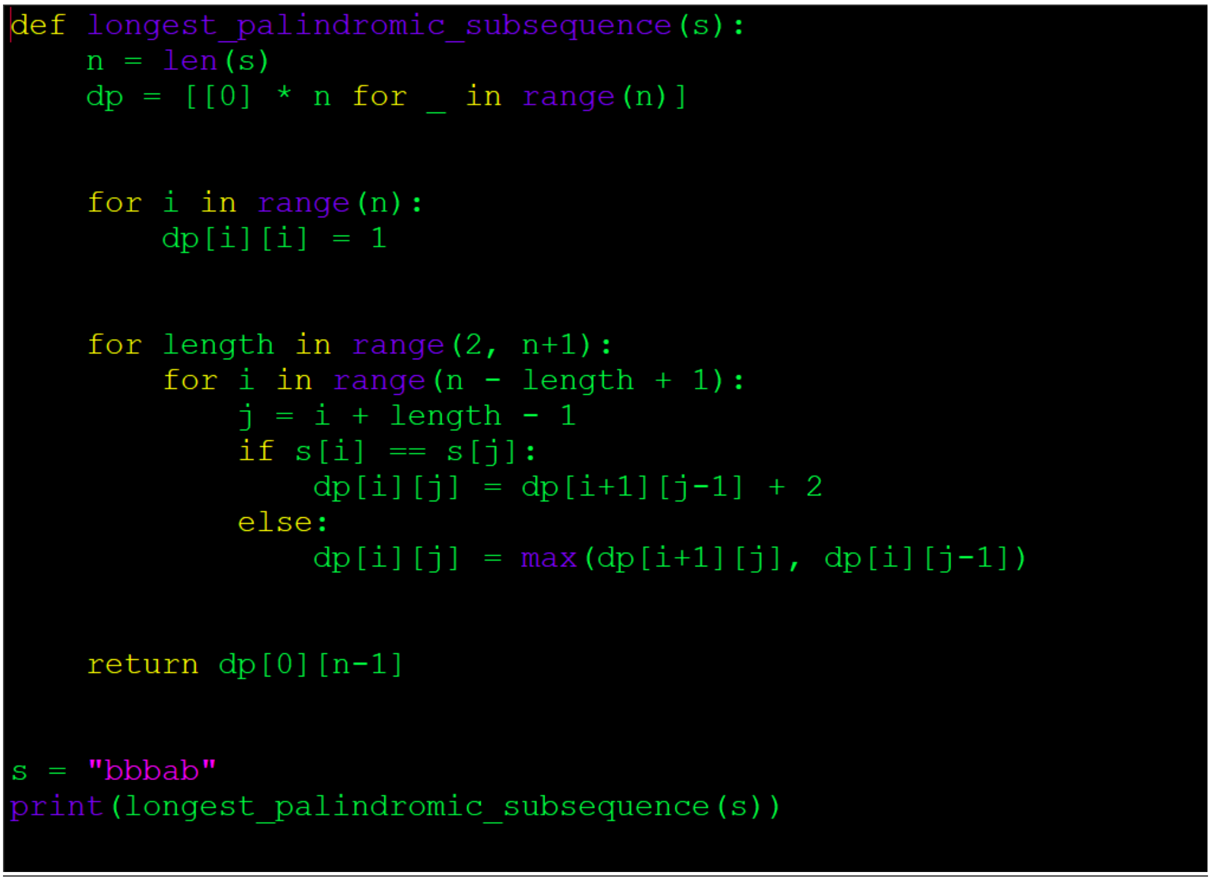
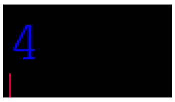
103. Longest palindromic subsequence

Code:



```
def longest_palindromic_subsequence(s):
    n = len(s)
    dp = [[0] * n for _ in range(n)]


    for i in range(n):
        dp[i][i] = 1


    for length in range(2, n+1):
        for i in range(n - length + 1):
            j = i + length - 1
            if s[i] == s[j]:
                dp[i][j] = dp[i+1][j-1] + 2
            else:
                dp[i][j] = max(dp[i+1][j], dp[i][j-1])


    return dp[0][n-1]


s = "bbbab"
print(longest_palindromic_subsequence(s))
```

Output:



```
4
```

Time Complexity:

- T(n)= O(n^2)