Exercise 166:-
6. Given a string s, return the longest palindromic substring in S.

Program:-
```
def longest_palindromic_substring(s):
    n = len(s)
    if n == 0:
        return ""
    dp = [[False] * n for _ in range(n)]
    start = 0
    max_length = 1
    for i in range(n):
        dp[i][0] = True
    for i in range(n - 1):
        if s[i] == s[i + 1]:
            dp[i][i + 1] = True
            start = i
            max_length = 2
    for length in range(3, n + 1):
        for i in range(n - length + 1):
            j = i + length - 1
            if s[i] == s[j] and dp[i + 1][j - 1]:
                dp[i][j] = True
                start = i
                max_length = length
    return s[start:start + max_length]
s = "babad"
print(f"The longest palindromic substring is: {longest_palindromic_substring(s)}")
```
Output:-
```
C:\Users\afree\PycharmProjects\pythonProject\.venv\Scripts\python.exe C:\Users\afree\PycharmProjects\pythonProject\0.py
The longest palindromic substring is: b

Process finished with exit code 0
```
Time complexity:-$O(n^2)$