

```

//Felipe Braz Marques / 22.1.4030
#include <stdio.h>
#include <string.h>..

/*
Forneça um algoritmo, baseado em programação dinâmica, que recebe como entrada uma cadeia de caracteres
e retorne como saída o tamanho da maior subsequência palíndromo da cadeia fornecida como entrada.
*/

int maiorSubsequenciaPalindromo(const char* s) {
    int n = strlen(s);
    int dp[n][n];

    for (int i = 0; i < n; i++) {
        dp[i][i] = 1;
    }

    for (int tamanho = 2; tamanho <= n; tamanho++) {
        for (int i = 0; i < n - tamanho + 1; i++) {
            int j = i + tamanho - 1;

            if (s[i] == s[j]) {
                if (tamanho == 2) {
                    dp[i][j] = 2;
                } else {
                    dp[i][j] = dp[i + 1][j - 1] + 2;
                }
            } else {
                dp[i][j] = (dp[i + 1][j] > dp[i][j - 1]) ? dp[i + 1][j] : dp[i][j - 1];
            }
        }
    }

    return dp[0][n-1];
}

// Função principal para testar o algoritmo
int main() {
    const char* s = "arara";
    int resultado = maiorSubsequenciaPalindromo(s);
    printf("Tamanho da maior subsequência palíndromo: %d\n", resultado);
    return 0;
}

```

```
38 int main() {  
39     const char* s = "arara";  
40     int resultado = maiorSubsequenciaPalindromo(s);  
41     printf("Tamanho da maior subsequência palíndromo: %d\n", resultado);  
42     return 0;  
43 }  
44
```



input

Tamanho da maior subsequência palíndromo: 5

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