

## HATFD1005

### Longest Palindromic Substring

Write a program to find the longest palindromic substring in a given string without using any built-in substring or reverse functions. For example, for input "babad", the output should be "bab" or "aba". **Instructions:** Avoid using any string handling libraries. Implement a solution that checks all substrings manually.

#### code

```
#include <stdio.h>

#include <string.h>

void longpalindromestring(char *s, char *result)

{

    int n = strlen(s);

    if (n == 0)

    {

        printf("NoT contain Palindromic Substring\n");

        return;

    }

    int dp[n][n];

    for (int i = 0; i < n; i++)

    {

        for (int j = 0; j < n; j++)

        {

            dp[i][j] = 0;

        }

    }

}
```

```
int maxLength = 1;

int start = 0;

for (int i = 0; i < n; i++)

{

    dp[i][i] = 1;

}

for (int i = 0; i < n - 1; i++)

{

    if (s[i] == s[i + 1])

    {

        dp[i][i + 1] = 1;

        start = i;

        maxLength = 2;

    }

}

for (int length = 3; length <= n; length++)

{

    for (int i = 0; i <= n - length; i++)

    {

        int j = i + length - 1;

        if (s[i] == s[j] && dp[i + 1][j - 1])

        {

            dp[i][j] = 1;

            if (length > maxLength)

            {

                start = i;
```

```

        maxLength = length;

    } } }

    strncpy(result, s + start, maxLength);

    result[maxLength] = '\0';

}

int main() {

    char a[100];

    char result[100];

    printf("Enter a string: ");

    scanf("%s", a);

    longpalindromestring(a, result);

    printf("Longest Palindromic Substring: %s\n", result);

    return 0;

}

```

## Input

Enter a string: wewewefsdsf

Longest Palindromic Substring: wewew

Enter a string: malayalam

Longest Palindromic Substring: Malayalam

Enter a string: momos

Longest Palindromic Substring: mom