**Assignment – 5**

**GitHub Link:**

<https://github.com/mdshakebj/Advance_Algorithm_Assignment-5>

#include <iostream>

#include <string>

#include <vector>

using namespace std;

int longestCommonSubstring(string X, string Y) {

int m = X.length();

int n = Y.length();

int maxLength = 0; // to store length of longest common substring

// initialize a 2D vector to store lengths of longest common suffixes of substrings

vector<vector<int>> LCSuff(m + 1, vector<int>(n + 1, 0));

// build LCSuff[m+1][n+1] in bottom up fashion

for (int i = 1; i <= m; i++) {

for (int j = 1; j <= n; j++) {

if (X[i - 1] == Y[j - 1]) {

LCSuff[i][j] = LCSuff[i - 1][j - 1] + 1;

maxLength = max(maxLength, LCSuff[i][j]);

}

else {

LCSuff[i][j] = 0;

}

}

}

return maxLength;

}

int main() {

string X = "BABA";

string Y = "ABAB";

cout << "Length of longest common substring is: " << longestCommonSubstring(X, Y) << endl;

return 0;

}

