**DAA Practical No 5**

|  |  |
| --- | --- |
| Name | Dhruv Bajaj |
| Batch | A6-B1 |
| Roll no | 06 |

**Task 1:**

**Code:**

X = "AGCCCTAAGGGCTACCTAGCTT"

Y = "GACAGCCTACAAGCGTTAGCTTG"

m = len(X)

n = len(Y)

Z = []

D = []

for i in range(m + 1):

    row = []

    for j in range(n + 1):

        row.append(0)

    Z.append(row)

for i in range(m + 1):

    row = []

    for j in range(n + 1):

        row.append(0)

    D.append(row)

print("Initial Z Matrix:")

for row in Z:

    print(row)

for i in range(1, m + 1):

    for j in range(1, n + 1):

        if X[i - 1] == Y[j - 1]:

            Z[i][j] = Z[i - 1][j - 1] + 1

            D[i][j] = 'd'

        else:

            if Z[i - 1][j] >= Z[i][j - 1]:

                Z[i][j] = Z[i - 1][j]

                D[i][j] = 'u'

            else:

                Z[i][j] = Z[i][j - 1]

                D[i][j] = 's'

print("\nFilled Z Matrix (LCS Lengths):")

for row in Z:

    print(row)

print("\nDirection Matrix (u = up, s = side, d = diagonal):")

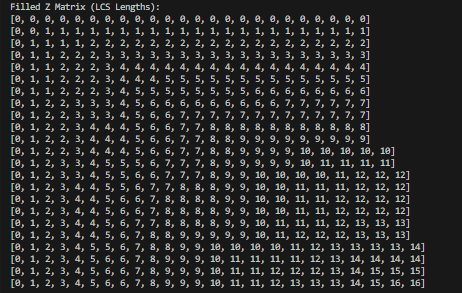
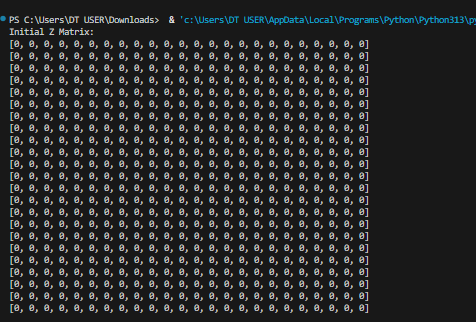
for row in D:

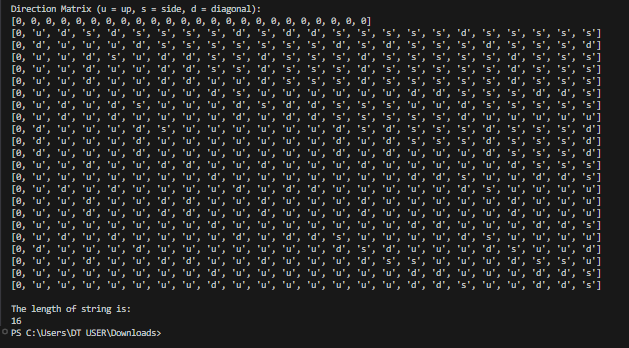
    print(row)

print("\nThe length of string is:")

print(Z[m][n])

**Output:**

****

****

**Task 2:**

**Code:**

def LRS(a, b):

    n = len(a)

    m = len(b)

    c = [[0] \* (m + 1) for \_ in range(n + 1)]

    for i in range(1, n + 1):

        for j in range(1, m + 1):

            if a[i - 1] == b[j - 1] and i != j:

                c[i][j] = 1 + c[i - 1][j - 1]

            else:

                c[i][j] = max(c[i - 1][j], c[i][j - 1])

    print(" Matrix:")

    for row in c:

        print(row)

    return c[n][m]

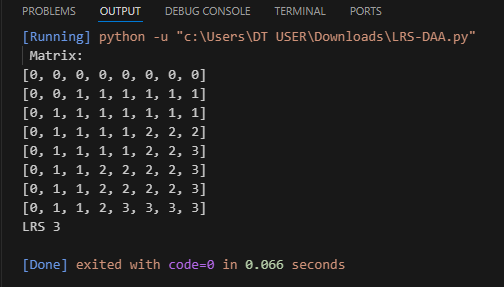
a = "AABCBDC"

b = "AABCBDC"

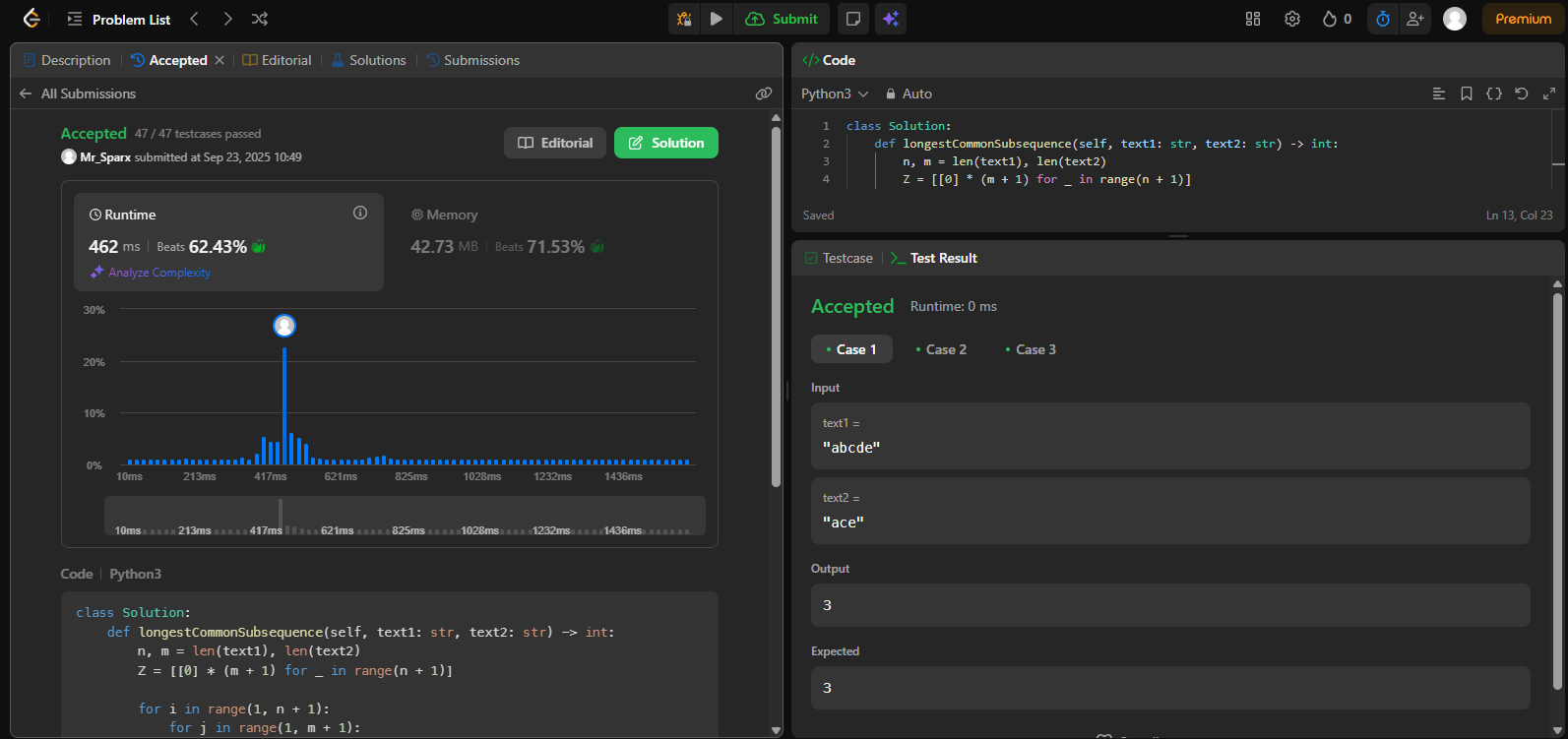
result = LRS(a, b)

print(f"LRS {result}")

**Output:**

****

**LeetCode:**

****