**Exercise – 7b**

**Binomial Coefficient**

**Aim:** To write a python code to implement Binomial Coefficient Algorithm

**Algorithm:**

* Divide and Conquer divides the problem of size C(n, k), in two sub problems, each of size C(n – 1, k – 1) and C(n – 1, k) respectively. Solution of larger problem is build by adding solution of these two sub problems.
* Structure of binomial coefficient problem using divide and conquer approach is described as :

C(n, k) = C(n-1, k-1) + C(n-1, k)

C(n, 0) = C(n, n) = 1

**Source Code:**

**1) Optimal Substructure:** follows the recursive structure

# A naive recursive Python implementation

def binomialCoeff(n, k):

    if k > n:

        return 0

    if k == 0 or k == n:

        return 1

    # Recursive Call

    return binomialCoeff(n-1, k-1) + binomialCoeff(n-1, k)

# Driver Program to test ht above function

n = 5

k = 2

print ("Value of C(%d,%d) is (%d)" % (n, k,

                                    binomialCoeff(n, k)))

**2) Overlapping Subproblems**

def binomialCoef(n, k):

    C = [[0 for x in range(k+1)] for x in range(n+1)]

    for i in range(n+1):

        for j in range(min(i, k)+1):

            if j == 0 or j == i:

                C[i][j] = 1

            else:

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

    return C[n][k]

n = 5

k = 2

print("Value of C[" + str(n) + "][" + str(k) + "] is "

    + str(binomialCoef(n, k)))

**Sample Input and Output:**

**1) Optimal Substructure:**

Input: n = 5

k = 2

Output:

A picture containing text, font, white, line

Description automatically generated

**2) Overlapping Subproblems**

Input: n = 5

k = 2

Output:

A picture containing text, font, white, line

Description automatically generated

**Result:**

Thus, Binomial Coefficient algorithm has been successfully implemented using Python code and the output is verified.