```
Inpunt: vector x, vector y
Output: matrix D, vector coefficient , string polynomial

begin  dividedDifferenceMethod

    n = size(x)
    D = matrix_zeros(n,n)

    D[:,0] = y.transpose()

    for i to n:
        aux0 = D[i-1:n,i-1]
        aux1 = adjacent_difference(aux0)
        aux2 = vector_subtraction(x[i:n],x[0:n-1-i+1])
        D[i:n,i] = vector_division(aux1,aux2.transpose())
    end

    coefficient = diagonal(D)

    polynomial = coefficient[0]
    m = '(x' + (-x[0]) + ')'
    for i to n:
        polynomial += coefficient[i] + m
        m += '(x' + -x[i] + ')'
    end
    D,coefficient ,polynomial

end dividedDifferenceMethod
```