Objective: **Implementation and analysis of Matrix multiplication**

Matrix Multiplication

The matrix multiplication can only be performed, if it satisfies this condition. Suppose two matrices are A and B, and their dimensions are A (m x n) and B (p x q) the resultant matrix can be found if and only if n = p. Then the order of the resultant matrix C will be (m x q).

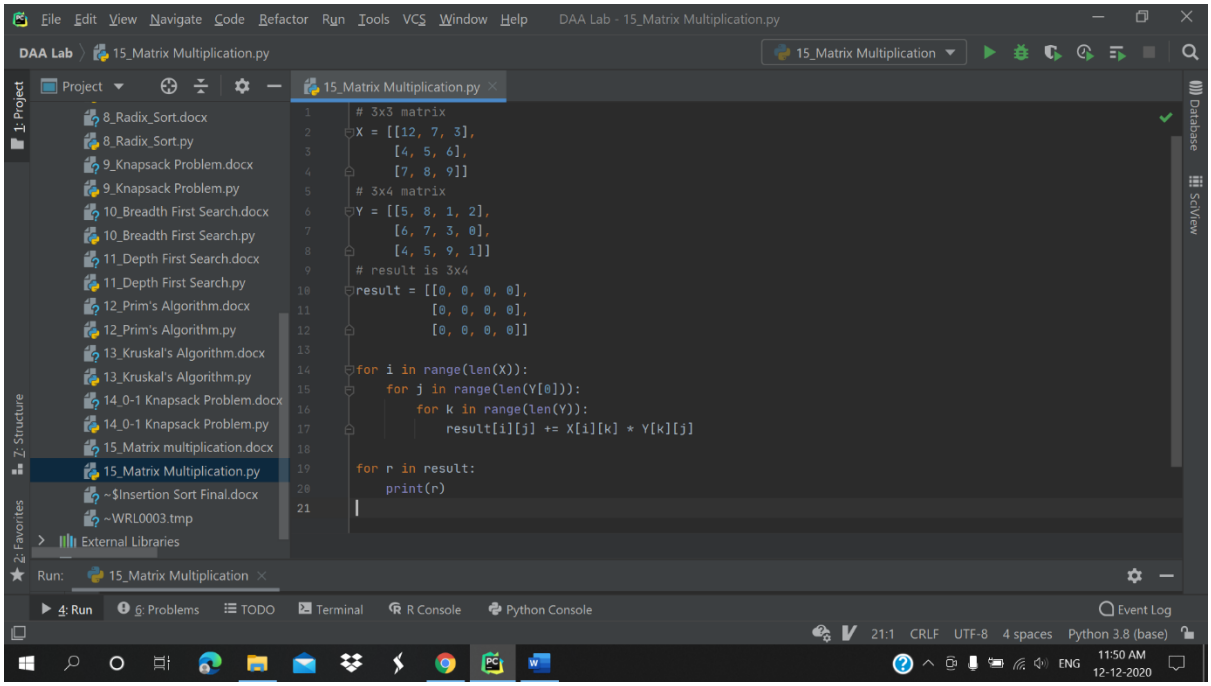
Algorithm:

```
matrixMultiply(A, B):
 Assume dimension of A is (m x n), dimension of B is (p x q)
Begin
        if n is not same as p, then exit
        otherwise define C matrix as (m x q)
        for i in range 0 to m - 1, do
                for j in range 0 to q – 1, do
                        for k in range 0 to p, do
                                C[i, j] = C[i, j] + (A[i, k] * A[k, j])
                        done
                done
        done
End
```

Code:

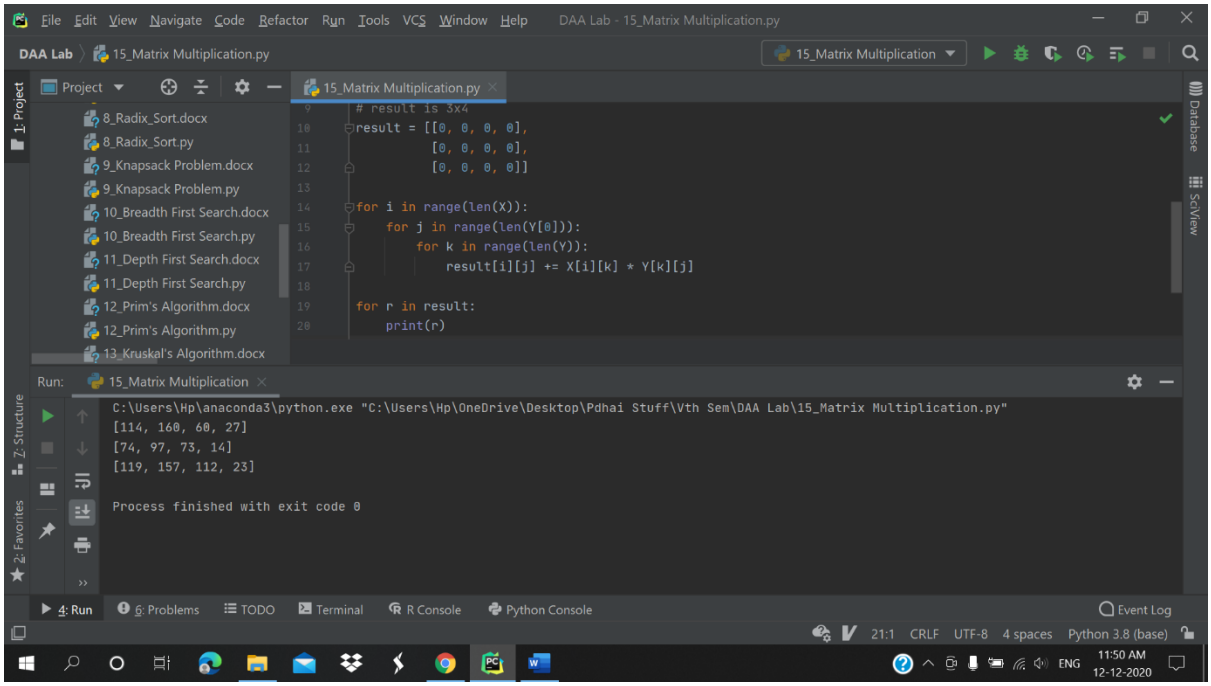
```
# 3x3 matrix
X = [[12, 7, 3],
    [4, 5, 6],
    [7, 8, 9]]
```

```
# 3x4 matrix
Y = [[5, 8, 1, 2],
     [6, 7, 3, 0],
     [4, 5, 9, 1]]
# result is 3x4
result = [[0, 0, 0, 0],
          [0, 0, 0, 0],
          [0, 0, 0, 0]]

for i in range(len(X)):
    for j in range(len(Y[0])):
        for k in range(len(Y)):
            result[i][j] += X[i][k] * Y[k][j]

for r in result:
    print(r)
```



Output:

```
[114, 160, 60, 27]
[74, 97, 73, 14]
[119, 157, 112, 23]
```



Time Complexity: The standard way of multiplying an m-by-n matrix by an n-by-p matrix has complexity O (mnp).