Data Analysis and Algorithm

Practical 3

Write a program on Strassen’s algorithm for matrix multiplication and analyze its

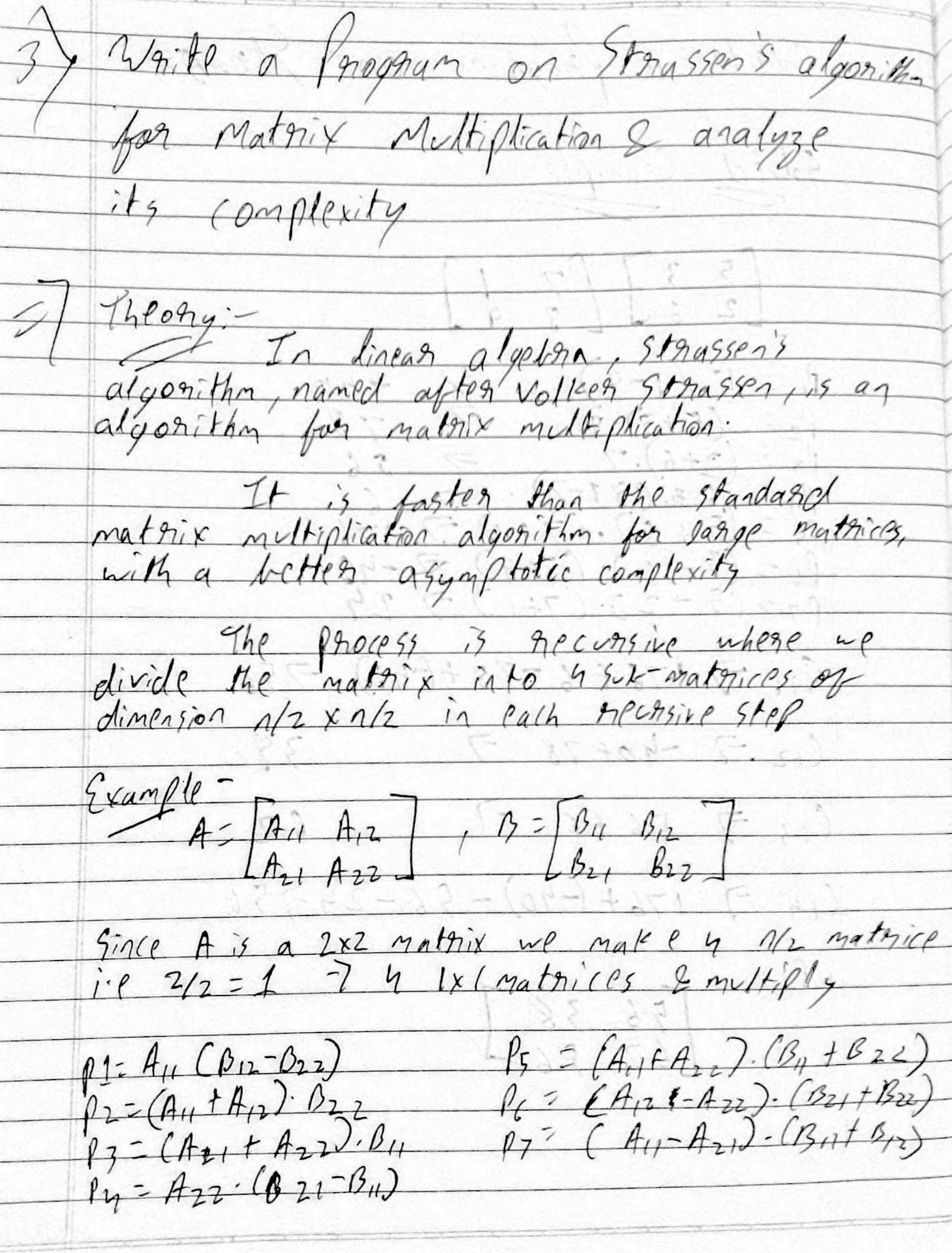
complexity.

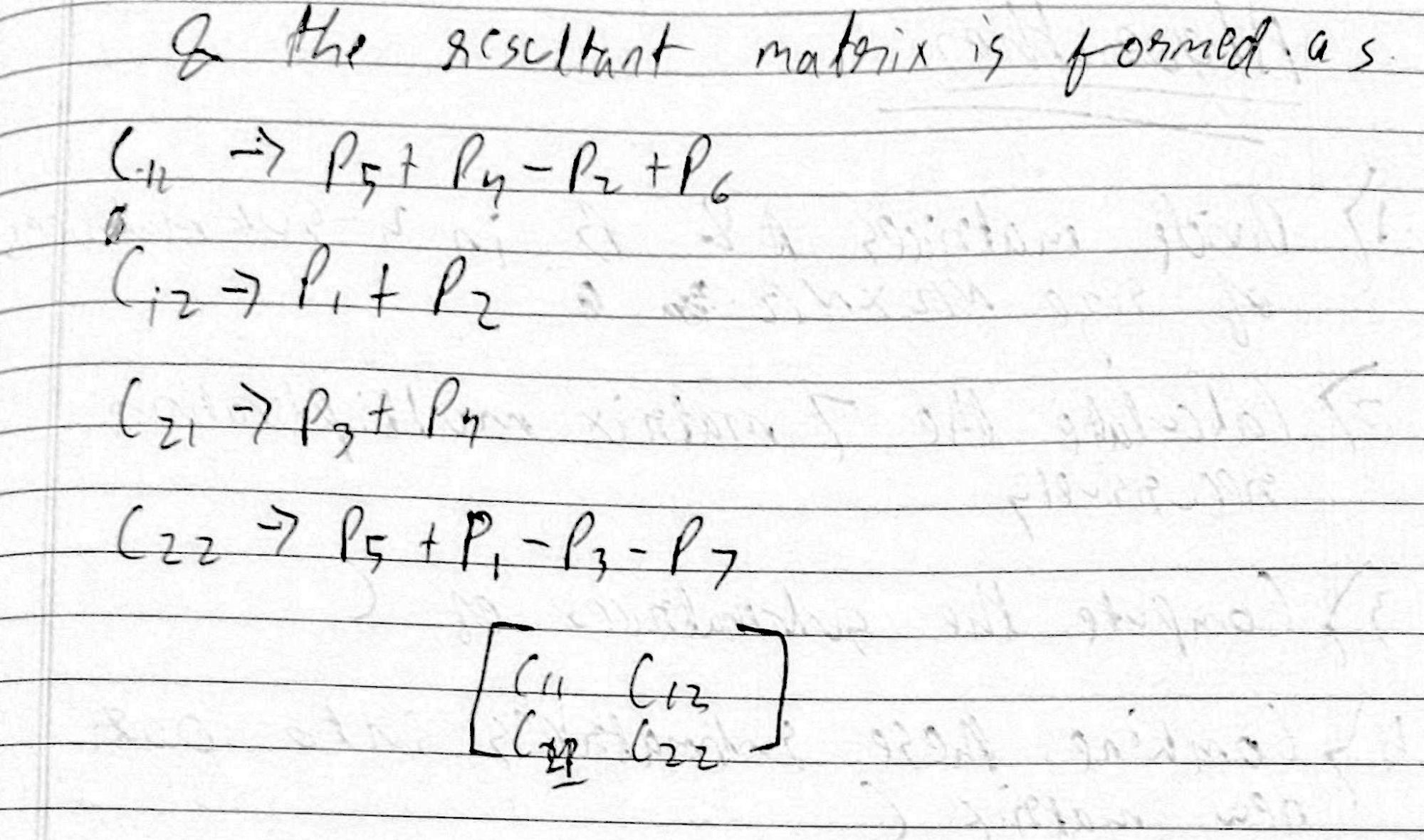
Date.: 15-09-21

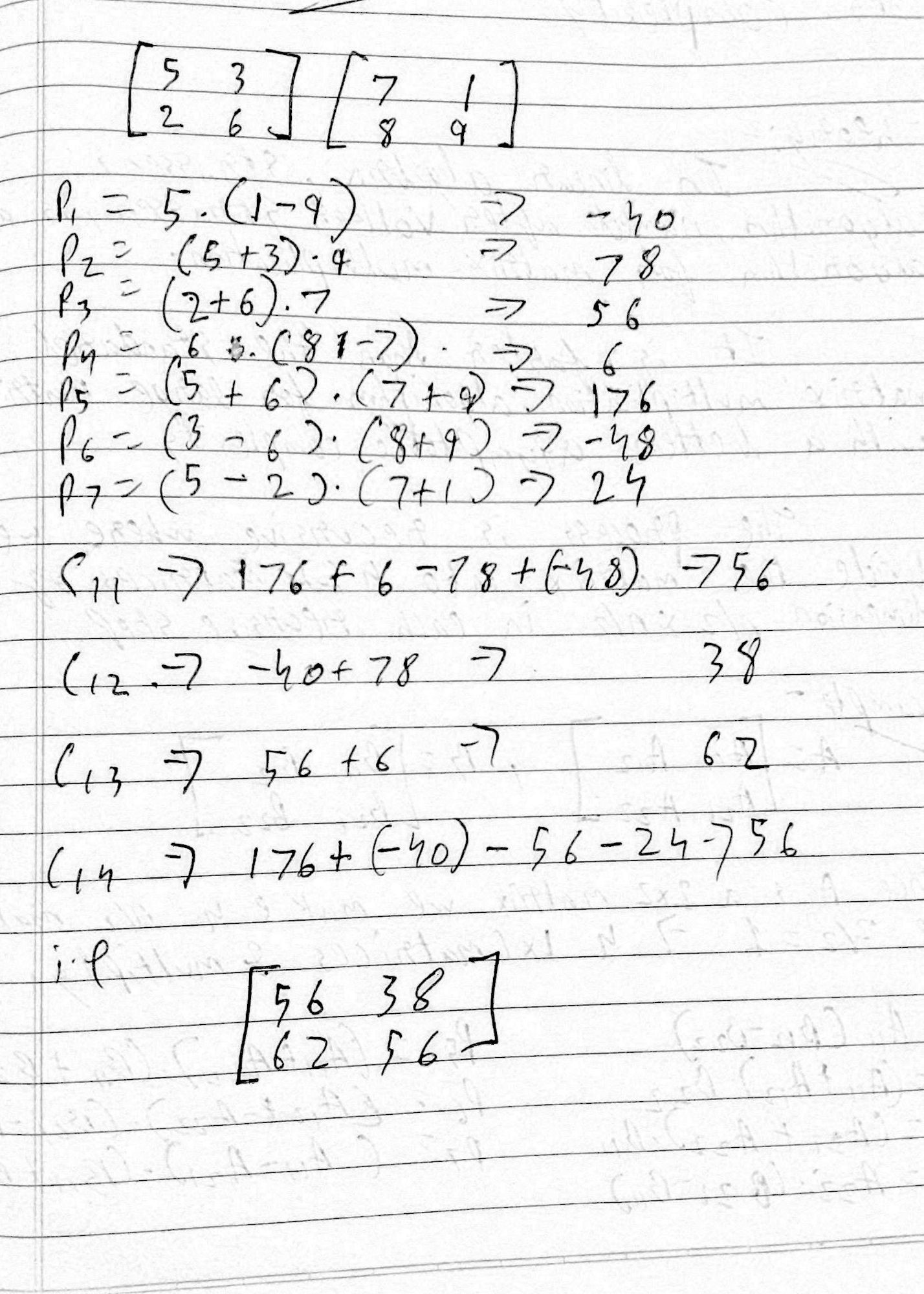
Name – Yash Vasudeo Prajapati

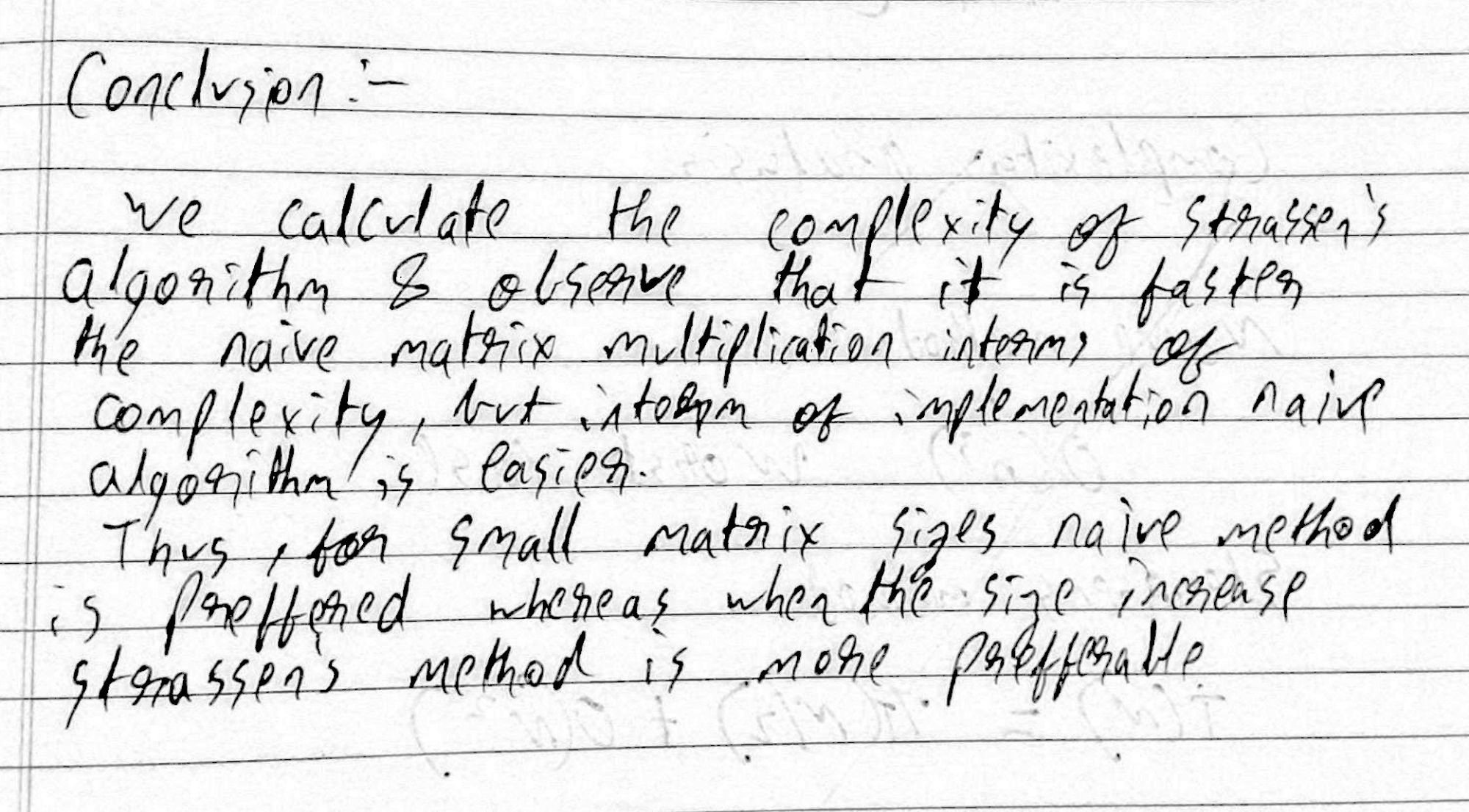
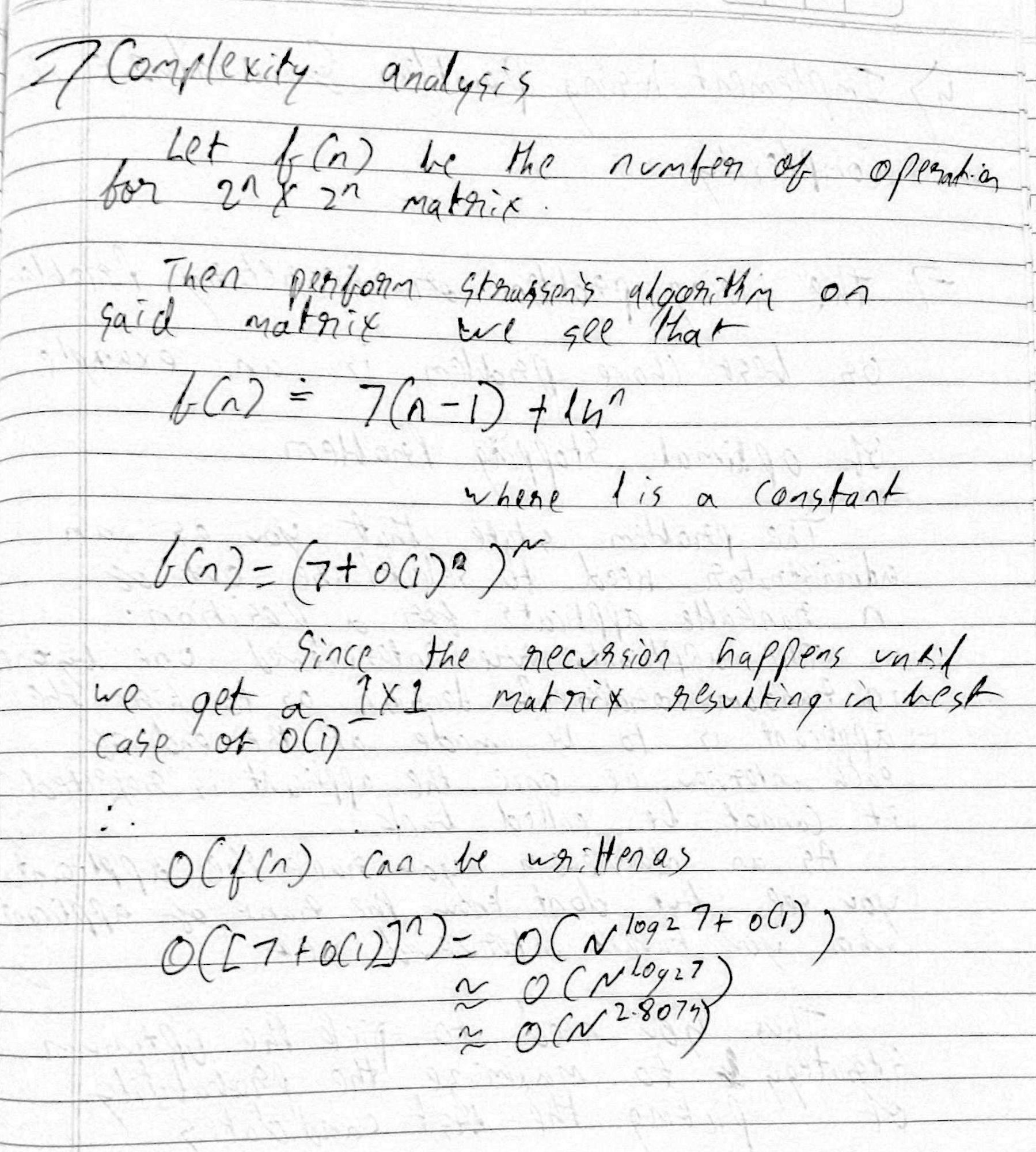
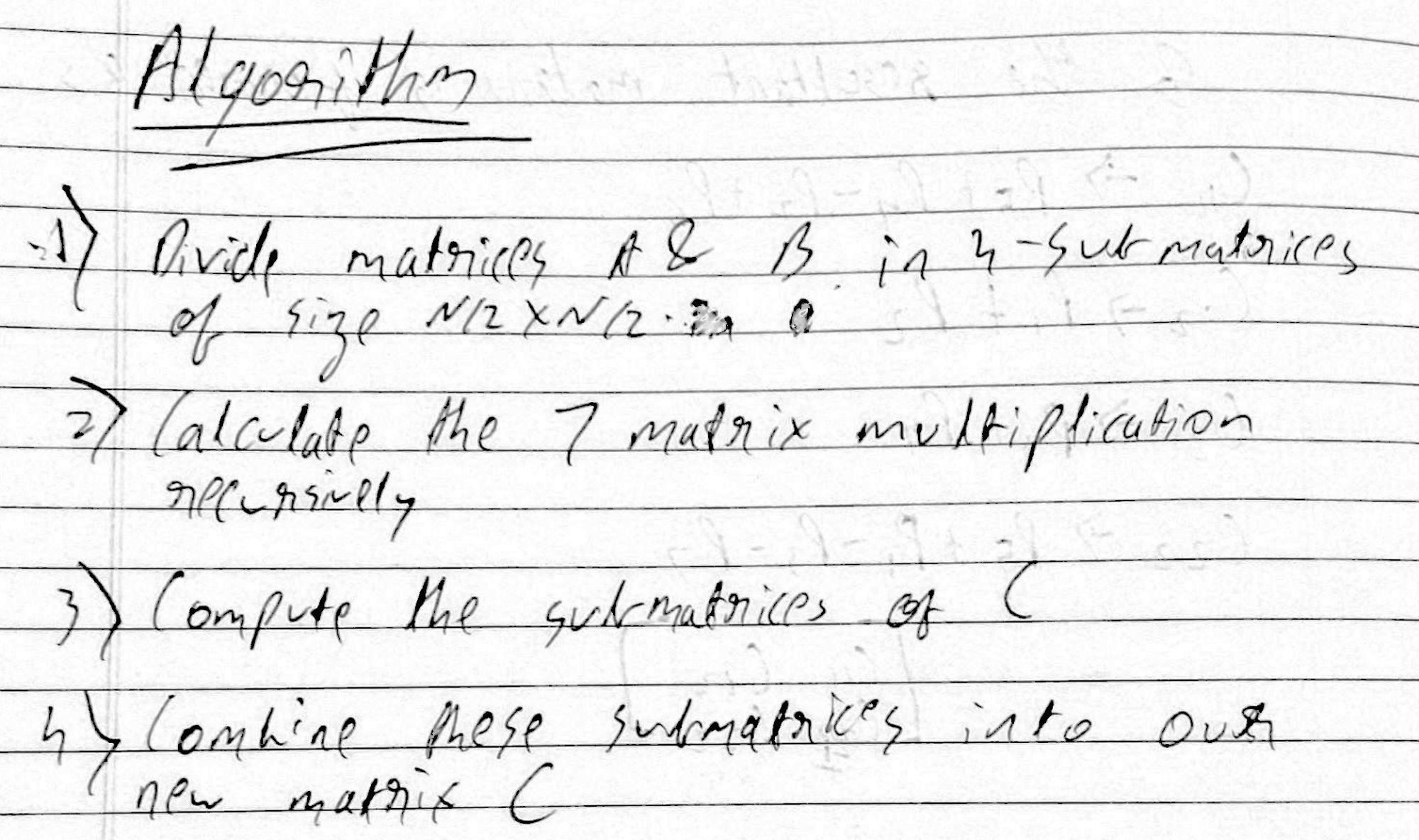
Rollno - 022

MSc. Computer Science









**Program :-**

1. import numpy as np
3. def split(matrix):
4. #spliting the matrix into n/2 size new matrices
5. row, col = matrix.shape
6. row2, col2 = row//2, col//2
7. return matrix[:row2, :col2], matrix[:row2, col2:], matrix[row2:, :col2], matrix[row2:, col2:]
9. def strassen(x, y):
10. # Base case when size of matrices is 1x1
11. if len(x) == 1:
12. return x \* y
14. # Splitting the matrices into quadrants
15. a, b, c, d = split(x)
16. e, f, g, h = split(y)
18. # Computing the 7 products, recursively (p1, p2...p7)
19. p1 = strassen(a, f - h)
20. p2 = strassen(a + b, h)
21. p3 = strassen(c + d, e)
22. p4 = strassen(d, g - e)
23. p5 = strassen(a + d, e + h)
24. p6 = strassen(b - d, g + h)
25. p7 = strassen(a - c, e + f)
27. # Computing the values of the 4 quadrants of the final matrix c
28. c11 = p5 + p4 - p2 + p6
29. c12 = p1 + p2
30. c21 = p3 + p4
31. c22 = p1 + p5 - p3 - p7
33. # Combining the 4 quadrants into a single matrix by stacking horizontally and vertically.
34. c = np.vstack((np.hstack((c11, c12)), np.hstack((c21, c22))))
36. return c
38. if \_\_name\_\_ == "\_\_main\_\_":
39. R = int(input("Enter the number of rows:"))
40. C = int(input("Enter the number of rows:"))
41. print("Matrix 1 values (separated by space): ")
42. inp1 = input()
43. print("Matrix 2 values (separated by space): ")
44. inp2 = input()
45. entries1 = list(map(int, inp1.split()))
46. entries2 = list(map(int, inp2.split()))
47. x = np.array(entries1).reshape(R, R)
48. y = np.array(entries2).reshape(R, R)
49. print(strassen(x,y))

