Data Analysis and Algorithm

Practical 2

Write a Program to implement Merge sort

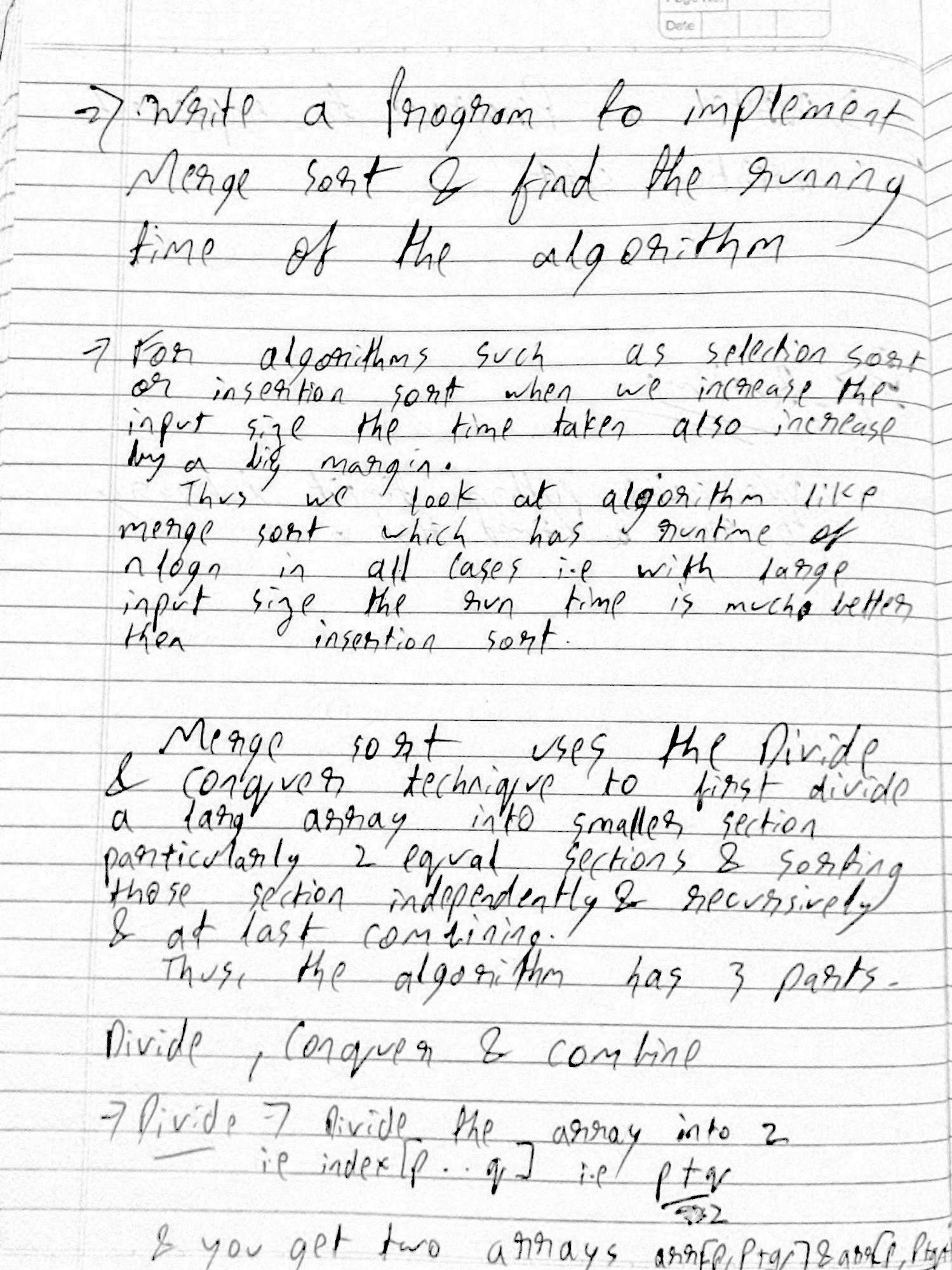
&

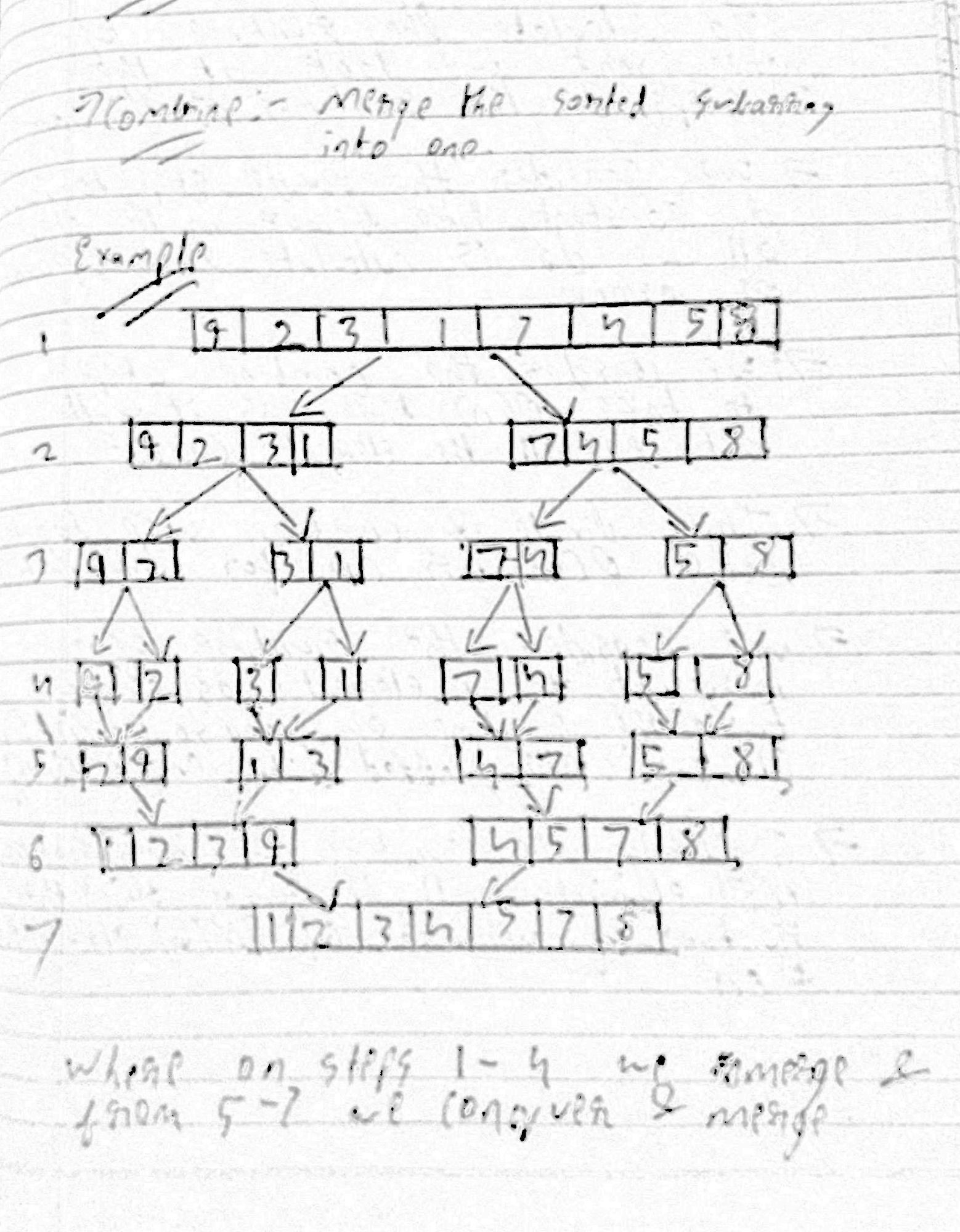
Find the run time of the algorithm

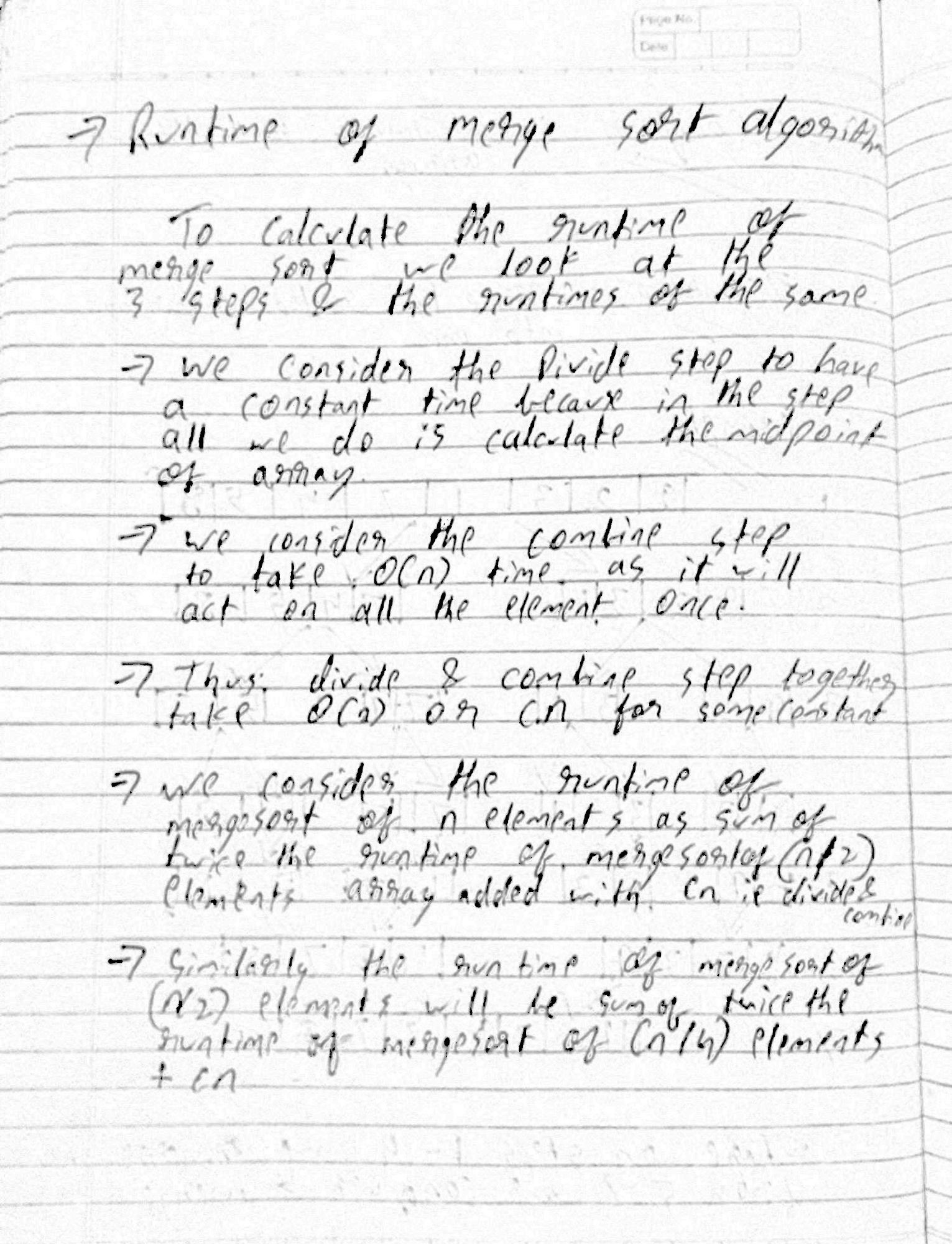
Name – Yash Vasudeo Prajapati

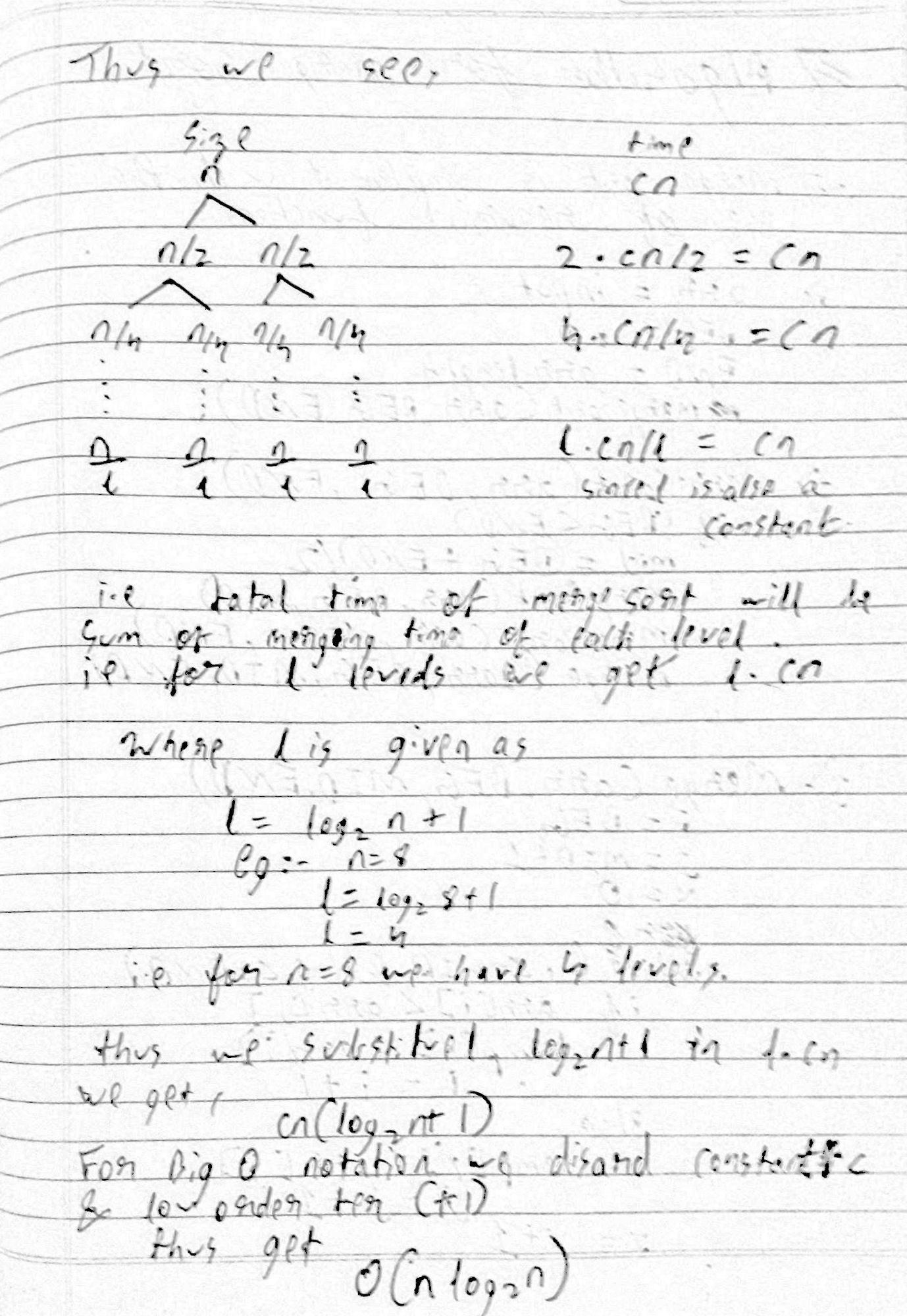
Rollno - 022

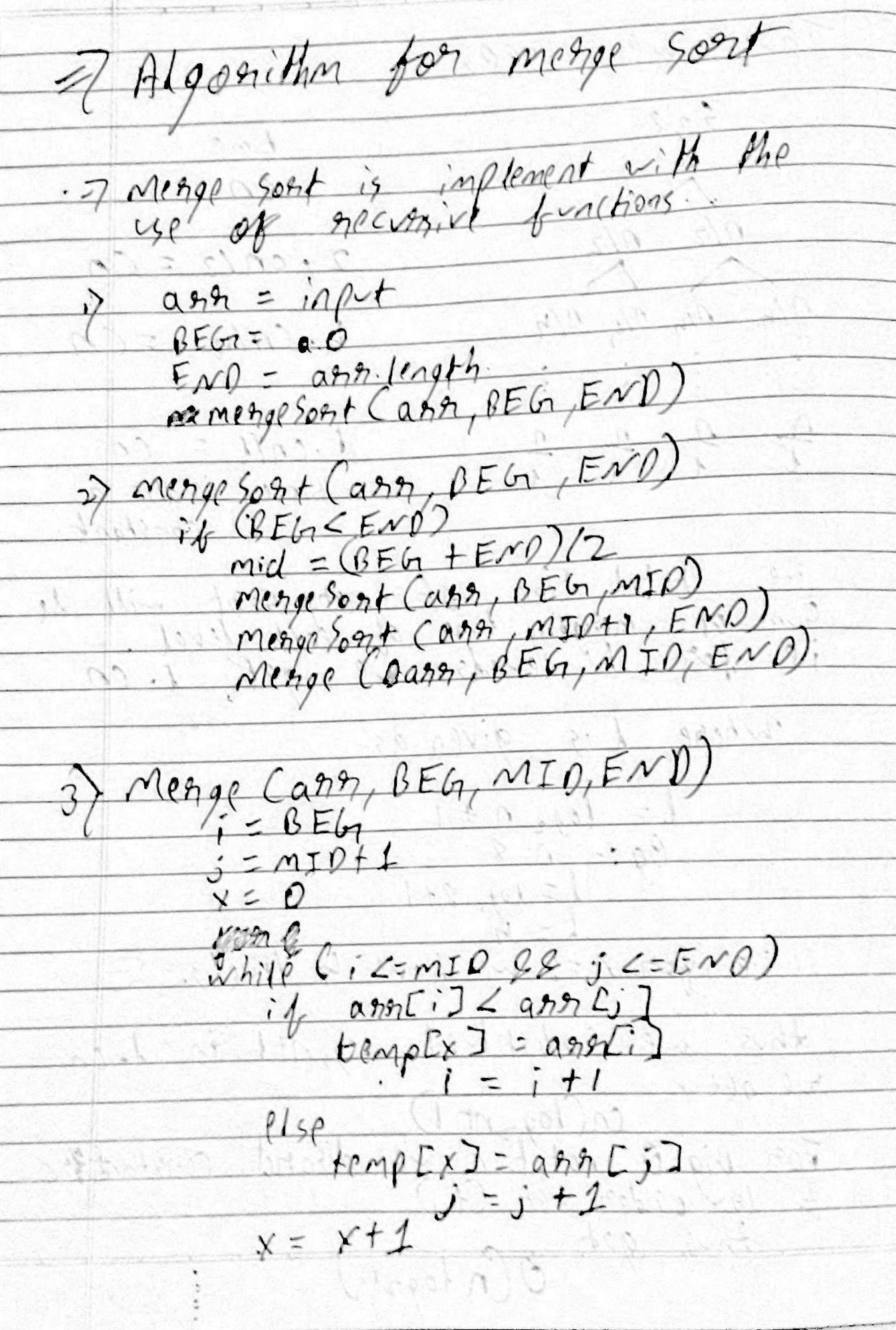
MSc. Computer Science

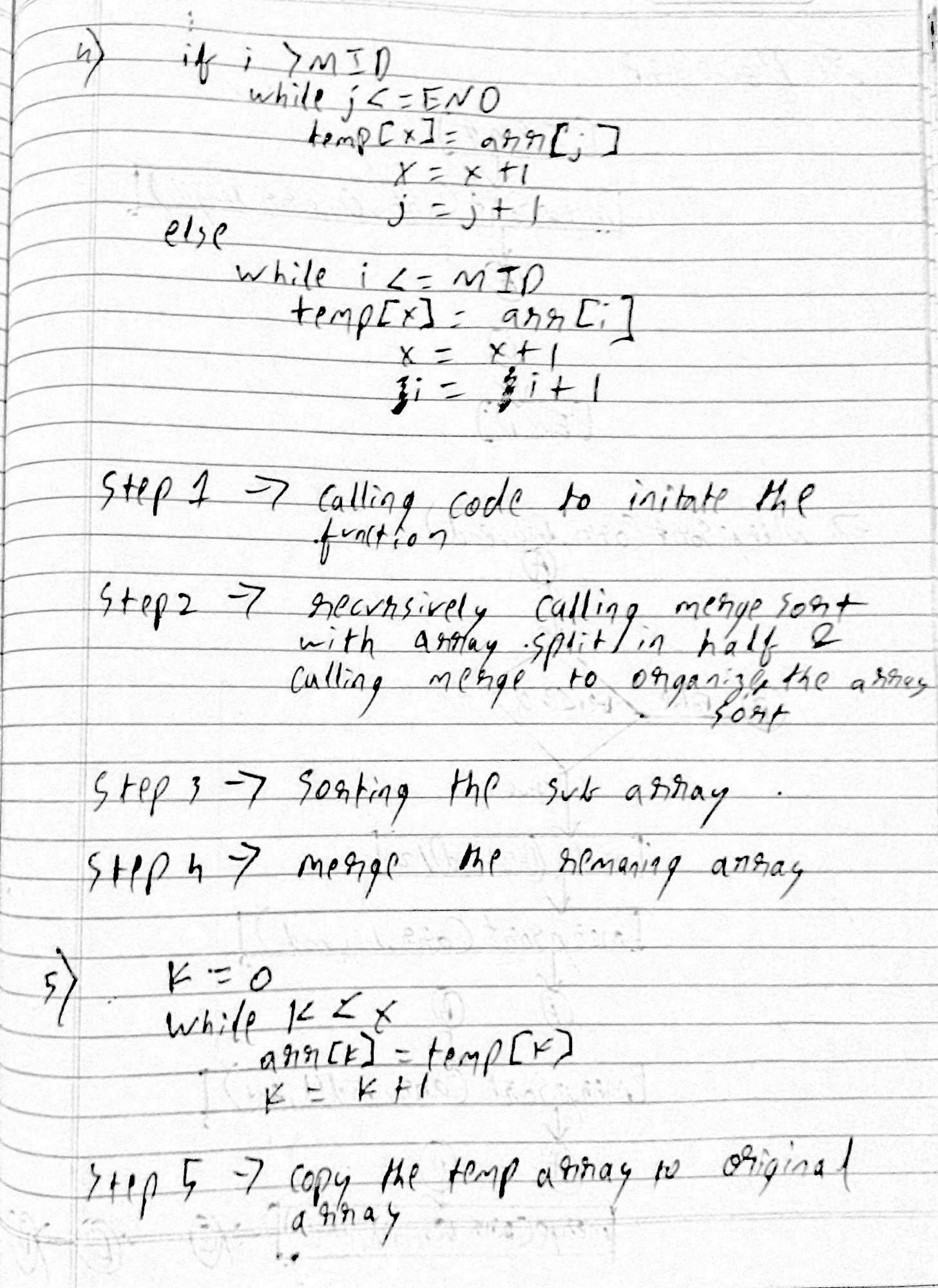


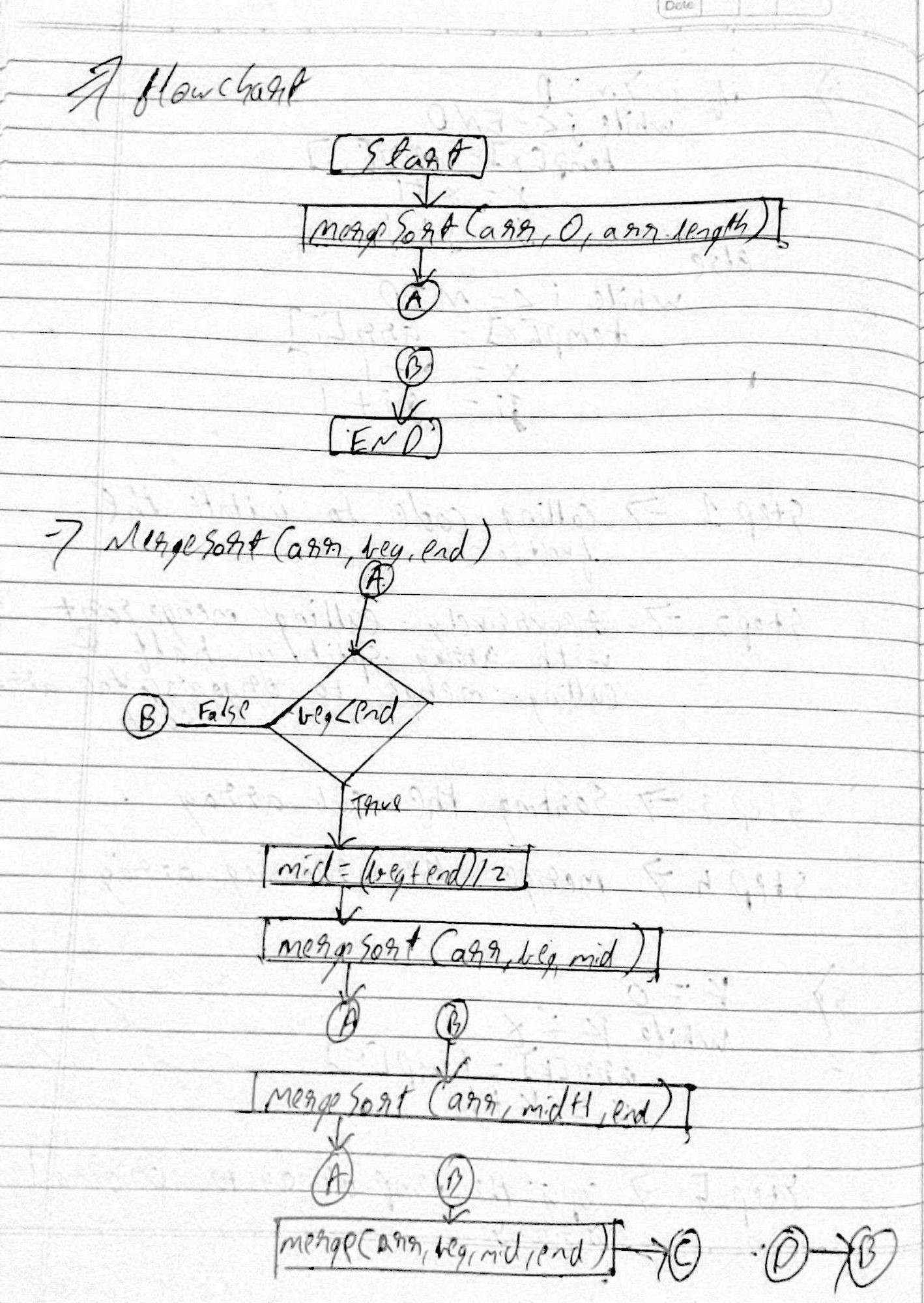


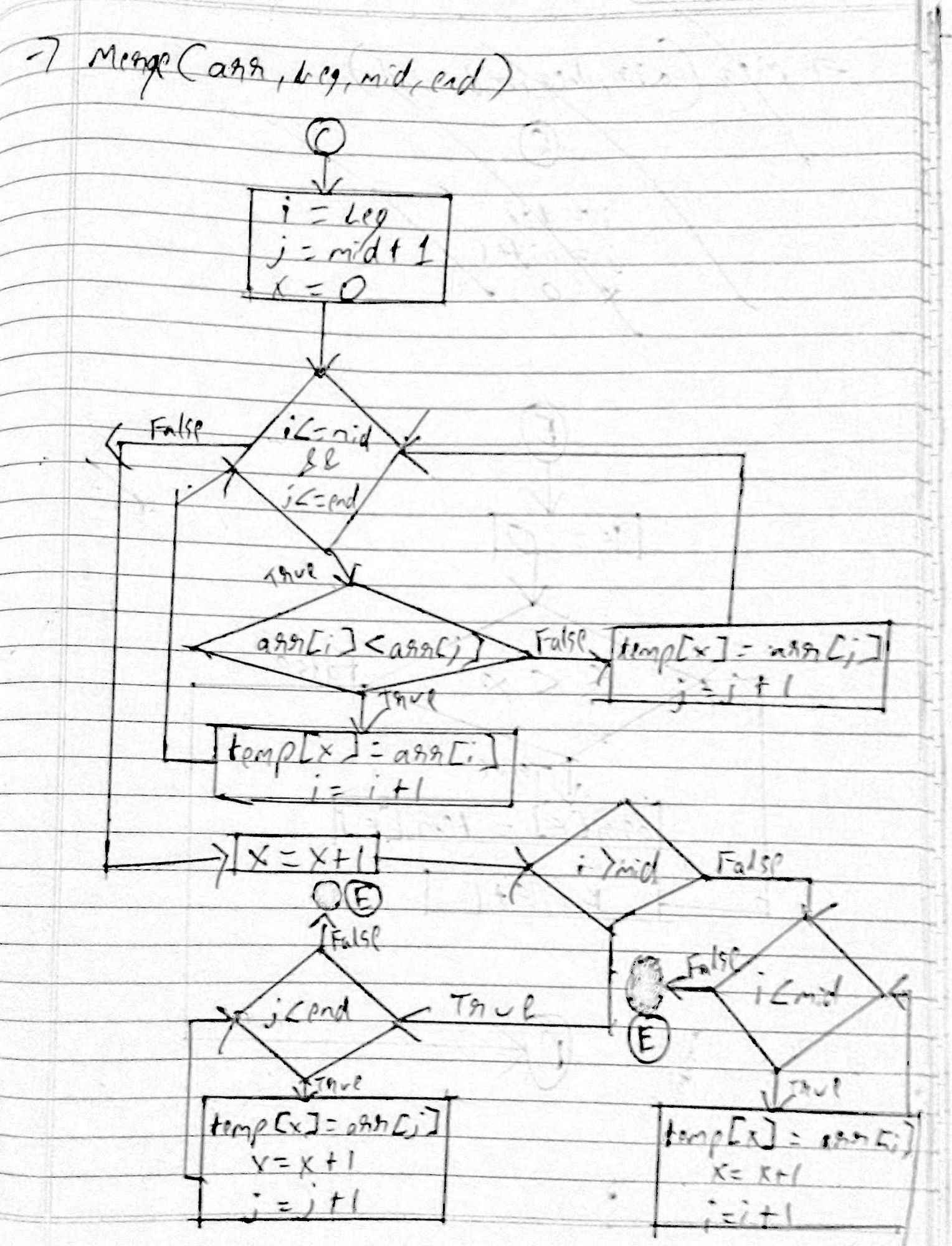


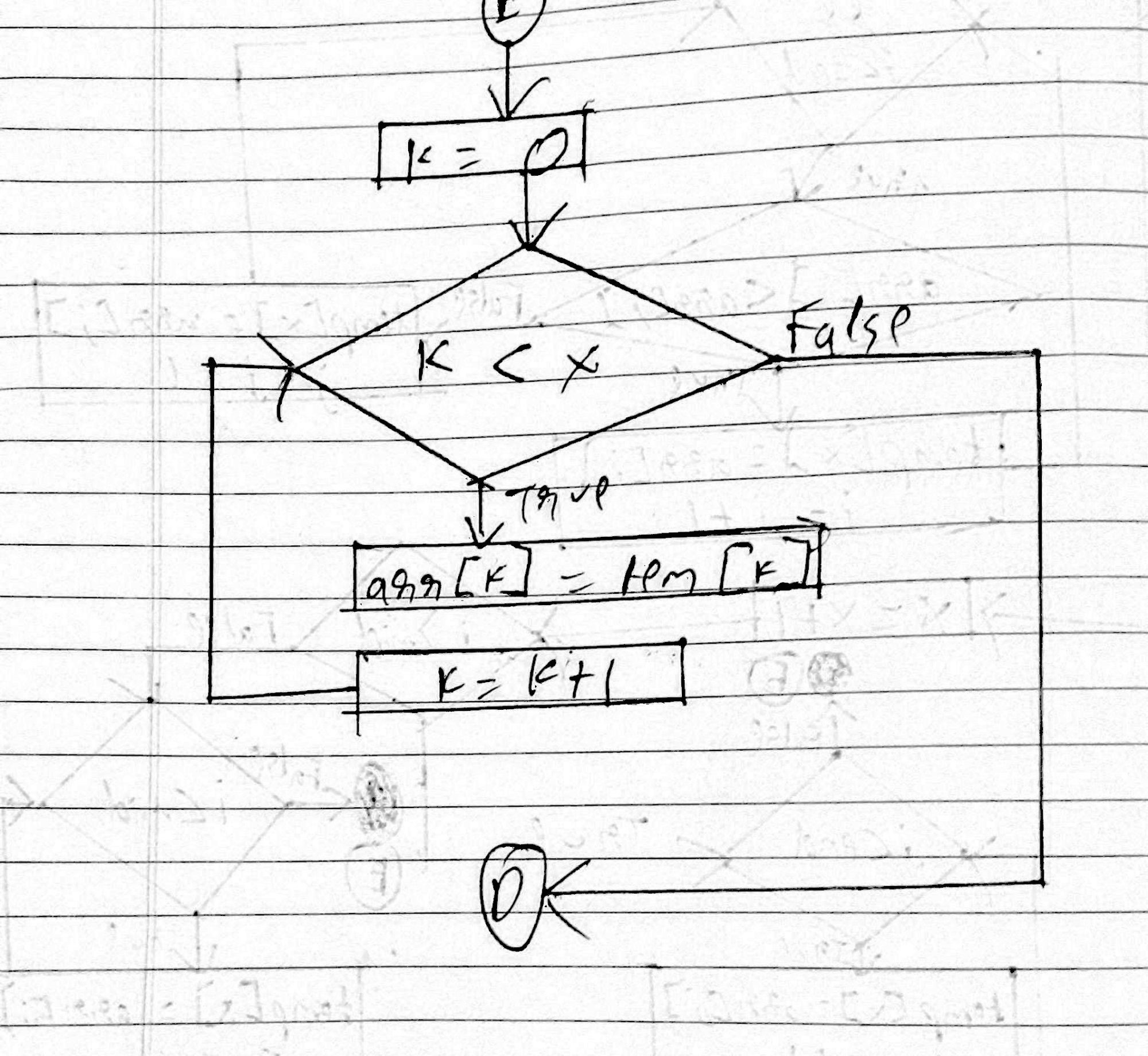


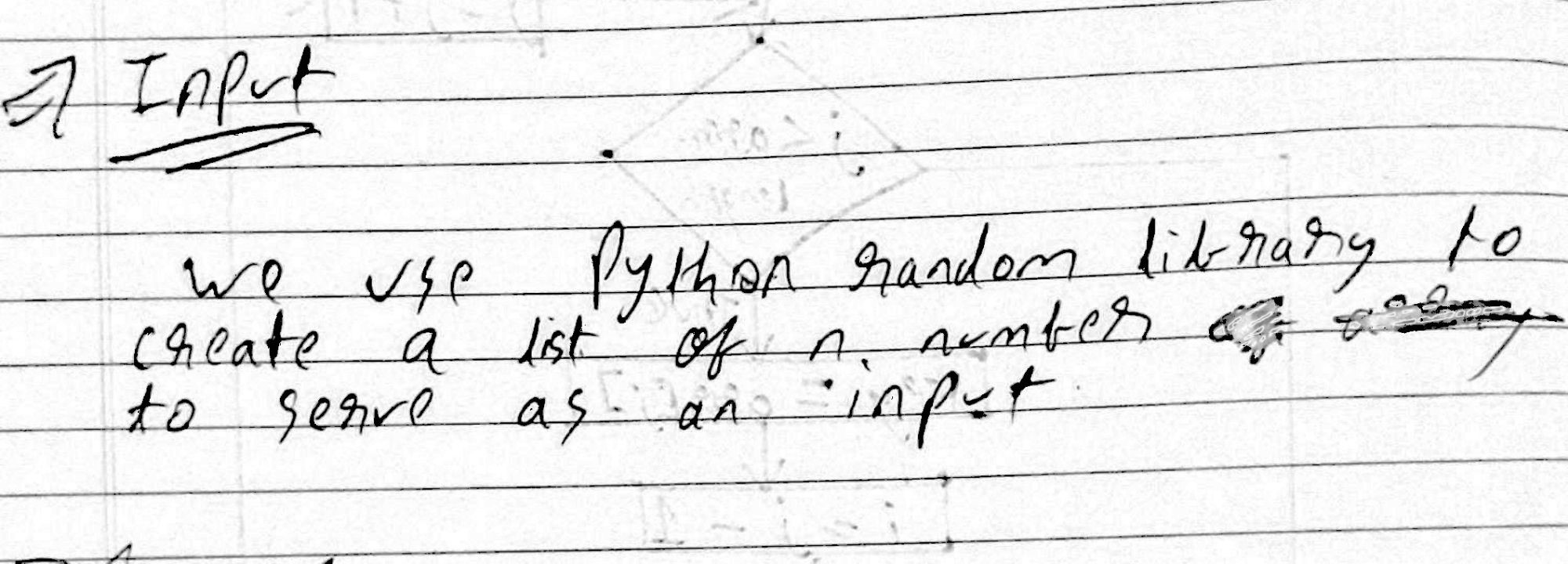


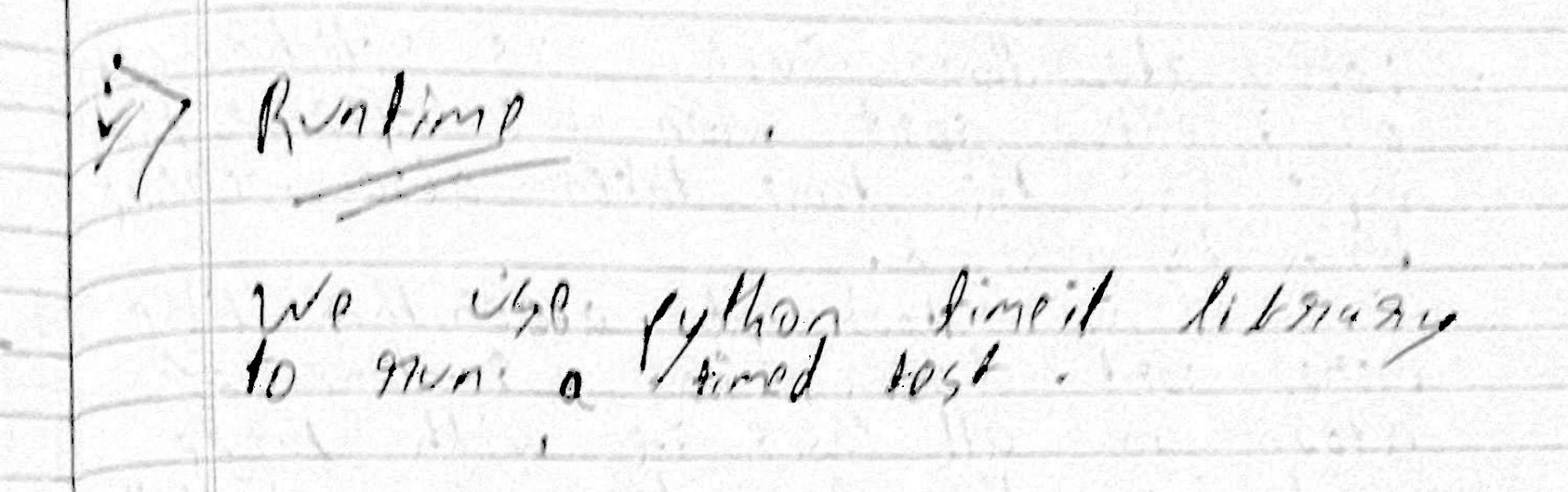






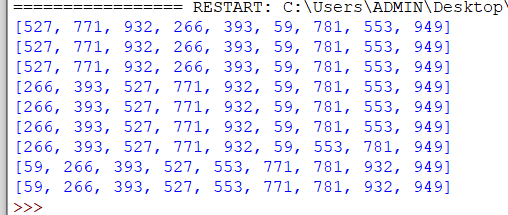






Program

1. import random
2. def merge(arr, l, m, r):
3. n1 = m - l + 1
4. n2 = r - m
6. # create temp arrays
7. L = [0] \* (n1)
8. R = [0] \* (n2)
10. # Copy data to temp arrays L[] and R[]
11. for i in range(0, n1):
12. L[i] = arr[l + i]
14. for j in range(0, n2):
15. R[j] = arr[m + 1 + j]
17. # Merge the temp arrays back into arr[l..r]
18. i = 0 # Initial index of first subarray
19. j = 0 # Initial index of second subarray
20. k = l # Initial index of merged subarray
22. while i < n1 and j < n2:
23. if L[i] <= R[j]:
24. arr[k] = L[i]
25. i += 1
26. else:
27. arr[k] = R[j]
28. j += 1
29. k += 1
31. # Copy the remaining elements of L[], if there
32. # are any
33. while i < n1:
34. arr[k] = L[i]
35. i += 1
36. k += 1
38. # Copy the remaining elements of R[], if there
39. # are any
40. while j < n2:
41. arr[k] = R[j]
42. j += 1
43. k += 1
45. # l is for left index and r is right index of the
46. # sub-array of arr to be sorted
47. def mergeSort(arr, l, r):
48. if l < r:
49. # Same as (l+r)//2, but avoids overflow for
50. # large l and h
51. m = l+(r-l)//2
52. # Sort first and second halves
53. mergeSort(arr, l, m)
54. mergeSort(arr, m+1, r)
55. merge(arr, l, m, r)
56. print(arr)
57. return(arr)
59. #driver code
60. if \_\_name\_\_ == "\_\_main\_\_":
61. arr=[]
62. for i in range (1,10):
63. n = random.randint(0,1000)
64. arr.append(n)
65. mergeSort(arr, 0, len(arr)-1)
66. print(mergeSort)



Timing code

1. SETUP\_CODE = '''
2. from \_\_main\_\_ import mergeSort,merge
3. import random
4. '''
6. TEST\_CODE = '''
7. arr=[]
8. input\_size = 10
9. for i in range (1,input\_size):
10. n = random.randint(0,1000)
11. arr.append(n)
12. mergeSort(arr, 0, len(arr)-1)'''
14. times = timeit.timeit(setup = SETUP\_CODE,
15. stmt = TEST\_CODE,
16. number = 1)
18. print(times)

Output time with array of size 10



Output time with array of size 100000



