DAA Lab 3

Page No.
Date

Aim → To understand how visual studio code works and its installation process, Code editing, productivity tips and learning about its Extensions. Writing an Algorithm to find gross and net salary of 2000 employees of ABC co. ltd. and find maximum and minimum salary using divide and conquer.

Input data → csv file having salary information of 2000 employees along with the breakup.

Theory → According to salary literature
- Gross salary → Basic salary + HRA + other Allowances
- Net salary → Gross salary - Professional tax - Employee Provident fund - Income tax.

Columns of csv file contain all these elements and row represents each Employee

Algorithm / Pseudocode

```
import pandas as pd
import numpy as np

df = pd.read_csv('Salary.csv')
df = np.array(df)
gross_salary =
basic_salary = df[basic_salary]
```

```
HRA = df["HRA"]
other = df['other_allowance']
prof_tax = df['profess
income_tax = df['income_tax']
employee_provident = df['employee_provid

gross_salary = basic_salary + HRA + other
net_salary = gross_salary - prof_tax -
             income_tax - employee_provi
```

- Linear Approach

```
min_salary = net_salary[0]
max_salary = net_salary[0]

for i = 0 to len(net_salary)-1:
    if (net_salary[i] > max_salary)
        max_salary = net_salary[i]
    if (net_salary[i] < min_salary)
        min_salary = net_salary[i]

print → max_salary
print → min_salary
```

→ Running time Complexity

1. Initialization → O(1)
2. for loop → $T_c = \sum_{i=1}^{n} 1 = n$

$\therefore T_c = O(n)$ [WorstCase]

3. If-conditions → O(1)
4. Print statement → O(1)

Page No.
Date

Divide and Conquer (Using Merge Sort)

def merge (arr[], left, mid, rio

Running time Complexity

1] Merge & J

Recursive Algo

findMinMax (arr[], left, right)

if (arr.size == 1)
return arr[0], arr[0]
else if (arr.size == 2)
if (arr[0] < arr[1])
return arr[0], arr[1]
else
return arr[1], arr[0]
else
mid = left + (right - left)//2
min1, max1 = findMinMax (arr[], left, mid)
min2, max2 = findMinMax (arr[], mid+1, right)

minimum = min (min1, min2)
maximum = max (max1, max2)

return maximum minimum

OR

Time Complexity
$T(n) = 2T(n/2) + O(n)$
Using Master Theorem
$T(n) = aT(n/b) + O(n^d)$
$a=2, b=2, d=1$ $a = b^d$ $\therefore T(n) = O(n)$

Test Cases

~~ID Basic Salary HRA Allowance Other Income Tax~~

1]

ID	Basic Salary	HRA Allowance	Other Income Tax	Prof. Tax	Provident fund
~~1~~	~~31403~~	~~18521~~	~~24370~~		
1	71422	31403	18521	24370	1428	8571
2	38795	6155	25979	15759	604	3695
3	15860	4691	9559	5040	1921	1903
4	53158	10019	21199	13134	1420	6379
5	77570	38402	12291	38339	314	9388
6	69343	33120	13160	6084	1876	8321
7	59732	6915	5112	19645	264	7168
8	26284	10047	26515	5486	1409	3154

Expected Output

Linear:

Min → ₹2367.00 (ID: 3)

Max → ₹101218.00 (ID: 6)

Divide and Conquer

Min → ₹23167.00 (ID: 3)

Max → ₹101218.00 (ID: 6)

2]

ID	Basic Salary	HRA Allowance	Other Income Tax	Prof. Tax	Provident fund
9	69486	32853	24581	31383	8386	1024
10	21765	3662	20924	9376	2552	2486
11	31850	10421	21523	8647	3822	1615
12	77618	36213	6424	32301	9321	1168
13	57491	5928	24333	4389	6263	568
14	36960	14007	24552	8056	4435	432
15	67191	13616	15808	17565	7463	2442
16	25[illegible]	[illegible]	[illegible]	[illegible]	9005	1106

Page No.
Date

Expected Output
Linear:
Min → ₹33923.00 (ID: 10)
Max → ₹96456.00 (ID: 16)
Divide and Conquer
Min → ₹33923.00 (ID: 10)
Max → ₹96456.00 (ID: 16)

3]

ID	Basic Salary	HRA	Other Allowance	Income Tax	Prof. Tax	Prof. Tax
17	59134	28881	11989	20260	7096	1170
18	75263	35954	28620	7835	9032	888
19	31023	8987	11261	4056	3723	376
20	56090	24950	5887	7188	6731	2464
21	16685	7305	24488	4718	2002	1234
22	79820	18170	21433	19685	9578	1669
23	15769	6805	25335	11404	1892	829
24	74735	8580	24944	29921	8968	363

Expected Output
Linear
Min → ₹34613 (ID: 23)
Max → ₹122970 (ID: 18)
Divide and Conquer
Min → ₹34613 (ID: 23)
Max → ₹122970 (ID: 18)

Page No.
Date

4]

ID	Basic Salary	HRA	Other Allowance	Income Tax	Prov Fund	Prof Tax
1	52976	20069	13053	10956	8151	452
2	28224	26425	15533	5880	9547	766
3	-50000	21935	6989	13209	4556	328
4	70445	20115	16081	11397	2404	351
5	59773		15142	14733	2287	1862
6	40433	8507	10567	14769	9181	1795
7	41264	5393	13520		9899	361
8	32238	7901	6111	7270	5442	1591

Expected Output:
Error: Negative Values found
Error: Empty Cells found in Column HRA
Error: Empty Cells found in Column Income Tax
Skipping Salary Calculation due to data Error.

5]

ID	Basic Salary	HRA	other Allowance	Income Tax	Prov fund	Prof Tax
1	55287	25738	6982	5464	8151	5389
2	76142	23632	14258	11338	9547	5674
3	-50000	28134	10726	14957	4576	8385
4	61211	14138	11292	3864	2404	3388
5	73450		19707	7932	2287	2426
6	32389	22169	11288	9742	9181	9910
7	67446	23307	14790		9899	6530
8	54286	26679	8297	6535	5442	2002

Expected output:
Error: Negative values found
Error: Empty Cells found
Error: Empty Cells found
Skipping Salary Calculations

Conclusion → Hence we have learnt the basics of Visual Studio code and code editing and its extensions. We created a CSV file having salary information of 2000 employees and found minimum and maximum salary by linear method having time complexity O(n) and Divide and Conquer using merge sort having time complexity of O(n log n)