DAA ASSIGNMENT 3

NAME:- Raunak Thanawala

Assignment:- Gross and Net Salary

Batch:- CE Group C

AIM:-

Write an algorithm to find gross and net salary of employees. ABC co. ltd. has 2000 employees.

Your task is to calculate each employee’s salary and find employee with minimum salary and maximum salary.

Do the above task using divide and conquer technique.

Find the improvement in the complexity using divide and conquer method.

CODE:-

"""

Name: Raunak Thanawala

Experiment: Gross and Net Salaries

Batch: C

Registration Number: 231070051

"""

import pandas as pd

class Salary:

    def \_\_init\_\_(self, input\_csv):

        self.input\_csv = input\_csv

        self.df = pd.read\_csv(self.input\_csv, delimiter=';')

        if pd.read\_csv(self.input\_csv, delimiter=';', nrows=1).empty:

            raise ValueError("EMPTY FILE")

        if self.df.isnull().any().any():

            raise ValueError("INVALID SIZE OF COLUMNS")

        if (self.df['Basic Salary'] < 0).any():

            raise ValueError("INVALID INPUT")

    def calculate\_gross\_salary(self):

        self.df['Gross Salary'] = (

            self.df['Basic Salary'] +

            self.df['House Rent Allowance'] +

            self.df['Other Allowances']

        )

    def calculate\_net\_salary(self):

        self.df['Net Salary'] = (

            self.df['Gross Salary'] -

            self.df['Income Tax'] -

            self.df['Provident Fund'] -

            self.df['Professional Tax']

        )

    def linear\_min\_max(self):

        net = self.df['Net Salary']

        ids = self.df['Sr.No']

        max\_salary = float("-inf")

        min\_salary = float('inf')

        max\_id = min\_id = 0

        for i in range(len(net)):

            if net.iloc[i] < min\_salary:

                min\_salary = net.iloc[i]

                min\_id = ids.iloc[i]

            if net.iloc[i] > max\_salary:

                max\_salary = net.iloc[i]

                max\_id = ids.iloc[i]

        return min\_id, max\_id

    def recursive\_min\_max(self, start, end):

        if start == end:

            employee\_id = self.df.iloc[start]['Sr.No']

            return employee\_id, employee\_id

        mid = (start + end) // 2

        min\_left\_id, max\_left\_id = self.recursive\_min\_max(start, mid)

        min\_right\_id, max\_right\_id = self.recursive\_min\_max(mid + 1, end)

        net\_min\_left = self.df[self.df['Sr.No'] == min\_left\_id]['Net Salary'].values[0]

        net\_min\_right = self.df[self.df['Sr.No'] == min\_right\_id]['Net Salary'].values[0]

        net\_max\_left = self.df[self.df['Sr.No'] == max\_left\_id]['Net Salary'].values[0]

        net\_max\_right = self.df[self.df['Sr.No'] == max\_right\_id]['Net Salary'].values[0]

        min\_id = min\_left\_id if net\_min\_left < net\_min\_right else min\_right\_id

        max\_id = max\_left\_id if net\_max\_left > net\_max\_right else max\_right\_id

        return min\_id, max\_id

def main():

    input\_csv = 'Salary5.csv'

    try:

        salary\_calculator = Salary(input\_csv)

        salary\_calculator.calculate\_gross\_salary()

        salary\_calculator.calculate\_net\_salary()

        min\_id\_linear, max\_id\_linear = salary\_calculator.linear\_min\_max()

        print("Linear:")

        print(f"ID with Minimum Salary: {min\_id\_linear}")

        print(f"ID with Maximum Salary: {max\_id\_linear}")

        min\_id\_recursive, max\_id\_recursive = salary\_calculator.recursive\_min\_max(0, len(salary\_calculator.df) - 1)

        print("Divide and Conquer:")

        print(f"ID with Minimum Salary: {min\_id\_recursive}")

        print(f"ID with Maximum Salary: {max\_id\_recursive}")

    except ValueError as e:

        print(f"Error: {e}")

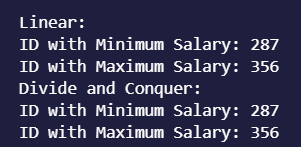
if \_\_name\_\_ == "\_\_main\_\_":

    main()

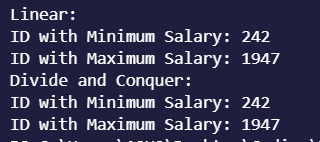
TESTCASES IN CODE :-

//INPUT: Inputs for TestCases are csv files provided with the docs

1.INPUT: Salary1.csv

OUTPUT: 

2.INPUT: Salary2.csv

OUTPUT: 

3.INPUT: Salary3.csv

OUTPUT: 

4.INPUT: Salary4.csv

OUTPUT: 

5.INPUT: Salary5.csv

OUTPUT: 

CONCLUSION :-

So In This Experiment we have Written Pseudo Code Algorithms for Gross and Net Salary with Input as a csv file and Output given outside of the algorithms.

We have also written 5 Test Cases where half of them are Positive and half of them are Negative. The testcases are given with input files as csv files and expected output written alongside it.

We have also calculated the Time Complexity of finding min and max Net Salary using Linear and Divide and Conquer way where we found that the Time Complexity of both methods is the same, O(n) with Master Method, This shows us that the Linear Way is the better method as the Divide and Conquer way is more complex to write and is more inefficient in space

Then we wrote a program following the pep8 style guide for python such that the ideal outputs were obtained in the program from the testcases.

We then executed the Program to give the Outputs for the given testcases.