Project Report:

Critical Thinking 1 - Calculate Average Withholding

Alejandro Ricciardi

Colorado State University Global

CSC320: Programming 1

Professor Herbert Pensado

May 5, 2024

Project Report:

Critical Thinking 1 - Calculate Average Withholding

This documentation is part of the Critical Thinking 2 Assignment from CSC320: Programming 1 at Colorado State University Global. This Project Report is an overview of the program's functionality, pseudocode, and detailed testing scenarios including console output screenshots. The program is coded in Java JDK-21; and is named Critical Thinking 2 (Calculate Average Withholding). The program is composed of a Main class and a TaxCalculator class.

The Assignment Direction

Create a program that will calculate the weekly average tax withholding for a customer given the following weekly income guidelines:
- Income less than $500: tax rate 10%
- Incomes greater than/equal to $500 and less than $1500: tax rate 15%
- Incomes greater than/equal to $1500 and less than $2500: tax rate 20%
- Incomes greater than/equal to $2500: tax rate 30%

Compile and submit your pseudocode, source code, and screenshots of the application executing the application, the results and GIT repository in a single document.

For source code please see Main.java and TaxCalculator.java files.

Git Repository

I use GitHub as my Distributed Version Control System (DVCS), the following is a link to my GitHub, Omegapy.
My GitHub repository that is used the store this assignment is named My-Academics-Portfolio and the link to this specific assignment is:
https://github.com/Omegapy/My-Academics-Portfolio/tree/main/Programming-1-CSC320/Critical-Thinking-2

Pseudocode

```
Program: Calculate Average Withholding
Author: Alejandro Ricciardi
Date: 05/05/2024

Description:
This program calculates the tax rate, withholding amount, and net income based on the weekly
income provided by the user. It repeatedly prompts for an income until the user decides to quit.

Begin Program

--------------------------------
File: TaxCalculator.java
--------------------------------
Class TaxCalculator
    // Methods to calculate tax-related values based on income brackets.

    Function calculateTaxRate(income)
        If income < 500 Then
            Return 0.10  // 10% tax rate
        ElseIf income >= 500 And income < 1500 Then
            Return 0.15  // 15% tax rate
        ElseIf income >= 1500 And income < 2500 Then
            Return 0.20  // 20% tax rate
        Else
            Return 0.30  // 30% tax rate
        End If
    End Function

    Function calculateWithholdingAmount(income)
        taxRate = calculateTaxRate(income)
        Return income * taxRate
    End Function

    Function calculateNetIncome(income)
        withholdingAmount = calculateWithholdingAmount(income)
        Return income - withholdingAmount
    End Function
End Class

--------------------------------
File: Main.java
--------------------------------
Class Main
    Main Program
        // Display program banner
        Print banner

        Create scanner object for input
        Loop  // Main loop to allow repeated calculations
```

```
            // Prompt and read income, validate format
            Loop
                Print "Please enter your weekly income: "
                strIncome = scanner.nextLine()
                If strIncome matches regex pattern for valid income Then
                    income = Convert strIncome to Double
                    Exit Loop
                Else
                    Print "Invalid input. Please enter a positive decimal number."
                End If
            End Loop

            // Calculate tax-related amounts
            taxRate = TaxCalculator.calculateTaxRate(income) * 100
            withholdingAmount = TaxCalculator.calculateWithholdingAmount(income)
            netIncome = TaxCalculator.calculateNetIncome(income)

            // Display results
            Print "---- Tax Withholding Summary ----"
            Print "Gross Income: $" + income
            Print "Withholding Percentage: " + taxRate + "%"
            Print "Amount Withheld: $" + withholdingAmount
            Print "Net Income: $" + netIncome

            // Prompt to continue or quit
            Print "Enter 'Q' to exit, or any other key to enter another weekly income: "
            quiteProgram = scanner.nextLine()
            If quiteProgram equals "q" (case insensitive) Then
                Print "Thank you for using the Calculate Average Withholding program!"
                Exit Main Loop
            End If
        End Loop
        Close scanner
    End Main Program
End Class

End Program
```

The TaxCalculator Class

The TaxCalculator class provides methods to calculate tax rates, withholding amounts, and net income based on an income brackets. This class is utilized by the Main class to perform detailed tax calculations based on inputted incomes.
The class has 3 methods.

The method **calculateTaxRatetax(**double income**)** computes the tax rate based on income.
The tax rate is determined by the income brackets:
   - Less than $500: 10%
   - $500 to $1499.99: 15%
   - $1500 to $2499.99: 20%
   - $2500 and above: 30%
It takes the parameter income, that is the weekly income entered by the user.
And returns the tax rate as a decimal (e.g., 0.10 for 10%).

The method **calculateWithholdingAmount**(double income**)** calculates the amount of income to be withheld as tax.
This method uses the tax rate determined by the calculateTaxRate method.
It takes the parameter income, that is the weekly income entered by the user.
And returns the amount of tax withheld.

The method **calculateNetIncome**(double income**)** calculates the net income after tax has been withheld by deducting the calculated withholding amount from the gross income.
It takes the parameter income, that is the weekly income entered by the user, aka the gross.
And returns the net income, the income after taxes.

The Main Class

The Main class runs the program, it calculates the taxes and displays the results based on the user-entered income.
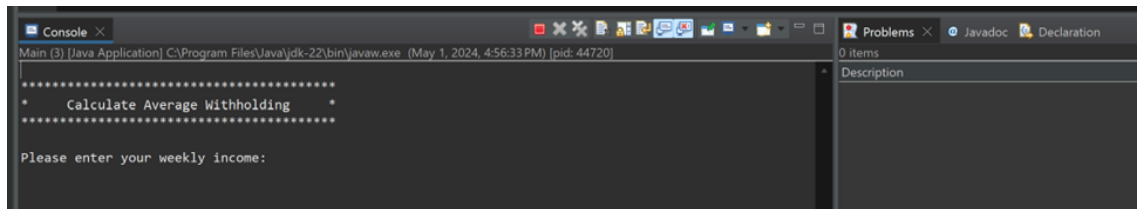It prompts the user for their weekly income, validates it, and calculates tax values.
It uses the TaxCalculator class to determine taxes.
It repeatedly prompts for user to enter an income until the user decides to quit.

Screenshots: Program Functionality and Testing Scenarios

Figure 1

Welcome Screen

Note: Eclipse Console output from the program and Problem window showing no items.

Figure 2

Validation Tests

Figure 2

Calculation Tests

```
Console ✕

<terminated> Main (3) [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (May 1, 2024, 4:56:33 PM – 5:00:24 PM) [pid: 44720]
*****************************************
*      Calculate Average Withholding      *
*****************************************

Please enter your weekly income: 100.00

---- Tax Withholding Summary ----
Gross Income: $100.00
Withholding Percentage: 10%
Amount Withheld: $10.00
Net Income: $90.00

Enter 'Q' to exit, or any other key to enter another weekly income: c


--------------------------------------------------------------------------------

Please enter your weekly income: 500.00

---- Tax Withholding Summary ----
Gross Income: $500.00
Withholding Percentage: 15%
Amount Withheld: $75.00
Net Income: $425.00

Enter 'Q' to exit, or any other key to enter another weekly income: c


--------------------------------------------------------------------------------

Please enter your weekly income: 1500.00

---- Tax Withholding Summary ----
Gross Income: $1500.00
Withholding Percentage: 20%
Amount Withheld: $300.00
Net Income: $1200.00

Enter 'Q' to exit, or any other key to enter another weekly income: c


--------------------------------------------------------------------------------

Please enter your weekly income: 2500.00

---- Tax Withholding Summary ----
Gross Income: $2500.00
Withholding Percentage: 30%
Amount Withheld: $750.00
Net Income: $1750.00

Enter 'Q' to exit, or any other key to enter another weekly income: q

Thank you for using the Calculate Average Withholding program!
```

As shown in Figure 1, Figure 2, and Figure 3 the test runs without any issues, displaying the correct output as expected.