A&DS 22-23 Formative Exercise

Task 1:

This task is to write an algorithm for three report outputs in pseudocode. I have selected the following three outputs, these are followed by the pseudocode:

Pseudocode Index:

Name	Action
Title	Indicates what action the algorithm will perform. Starts the algorithm
FOR	Begins a for loop.
END FOR	Ends for loop.
( )	Variables used within the algorithm.
IF ELSE	If else statement.
END IF	Ends the if statement.
PRINT	Opt to use Print instead of return, explained further below.
END	Ends the algorithm.

[1] [2]

Note I have made the following assumptions when writing:

1. As mentioned above, the decision was made to use the print statement rather than return. In introduction to Algorithms, it is stated that a "return statement immediately transfers control back to the point of call in the calling procedure." The print function was used to avoid this, as the use of the return function would close for loops when attempting to present the answers. *[2]*
2. Individuals attempting to read the Pseudocode, will first read the description of the algorithm.
3. quarterTotal has been assigned within the first algorithm, I will be using this for algorithms going forward. This has been additionally implicitly clarified within my Pseudocode.
4. Figures have been converted from £1000, to just £1 equivalent. Eg. 208, as £1000 is now displayed as £208000. This was done to avoid large decimals on the tax calculations.

A&DS 22-23 Formative Exercise

Algorithm 1

The total sales for each department per quarter i.e. “2nd Quarter totals: Electrical, £208,000”

Pseudocode: [2]

```
TOTAL SALES PER QUARTER (departmentSales, d, m, q)
        //departmentSales are those from table 1 from the assignment brief. This is a 2D array.
        // d = department, m = month, q = quarter
        // populate quarterTotal
        FOR d from 1 to 5
                FOR m from 1 to 6
                        IF m < 4, quarterTotal [d] [1]= quarterTotal [d] [1]  + departmentSales [d] [m]
                                ELSE quarterTotal [d] [2]= quarterTotal [d] [2]  + departmentSales [d] [m]
                         END IF
                END FOR
        END FOR

        // display quarterTotal
        FOR d from 1 to 5
                FOR q from 1 to 2
                        PRINT quarterTotal [d] [q]
                END FOR
        END FOR
END
```

A&DS 22-23 Formative Exercise

Algorithm 2

Given the total sales for each quarter, calculate the tax that needs to be paid at 17%.

Pseudocode: [2]

```
TAXOWED (quarterTotal, sumToBeTaxed, t, d, q )
        //quarterTotal taken from Algorithm 1
        // t = taxRate, d = department, q = quarter
        //set sumToBeTaxed
        FOR q from 1 to 2
                FOR d from 1 to 5
                        sumToBeTaxed = sumToBeTaxed + quarterTotal[q][d]
                END FOR
                PRINT sumToBeTaxed  * t
        END FOR
END
```

Algorithm 3

Given the average sales for each department across the last reported quarter, provide a new sales target for each department with an increase of 12%

Pseudocode: [2]

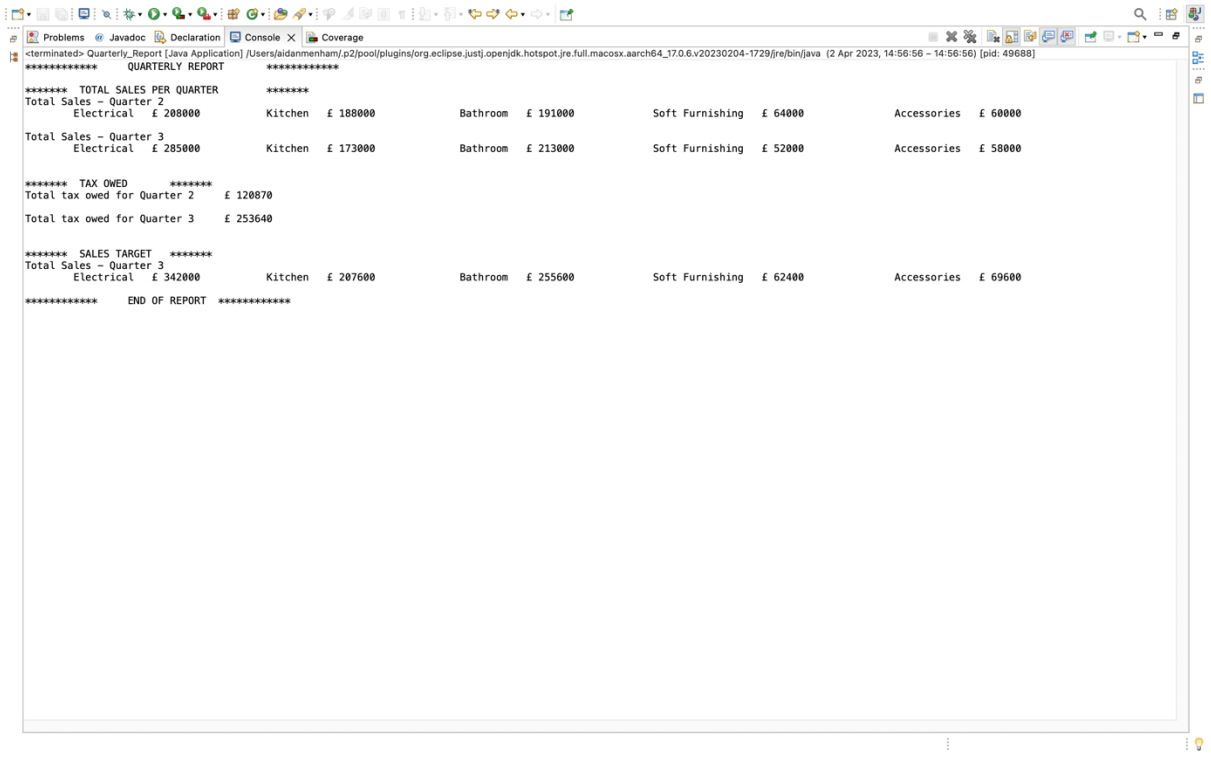
```
SALESTARGET (quarterTotal, sumToBeTaxed, t, d, q )
        //quarterTotal taken from Algorithm 1
        //t = target, d = department, q = quarter
        FOR q from 1 to 2
                FOR d from 1 to 5
                        totalSales = quarterTotal[q][d] * t
                END FOR
                PRINT totalSales
        END FOR
END
```

A&DS 22-23 Formative Exercise

Task 2:

This task is to implement the algorithms defined above in a single prototype application.

Prototype application attached, as part of this submission, results screenshotted and included below.



[3]

Bibliography

[1] K. S. University, “Kennesaw State University,” [Online]. Available: https://ccse.kennesaw.edu/fye/pseudocode/pseudocodeguide.php. [Accessed 18th March 2023].

[2] T. Corman and E. Al., Introductions to Algorithims - 4th Edition, Cambridge, Massachusetts: The MIT Press, 2022.

[3] Q. Charatan and A. Kans, Java in Two Semesters - Fourth Edition., London, UK: Springer Nature Switzerland, 2019.