7 A program is being developed to implement a customer loyalty scheme for a coffee shop.

The programmer has decided that the following data items need to be stored for each customer:

Data item	Description
customer ID	a unique six-digit string
loyalty points	an integer value that depends on how often the customer visits the coffee shop
last visit date	the date the customer last visited the coffee shop

The programmer has defined a program module:

Module	Description
`UpdateVisit()`	• called with two parameters: 1. loyalty points of type `INTEGER` 2. last visit date of type `DATE` • change the loyalty points: ○ increase by **four** if the last visit date and the current date are in the same month ○ otherwise increase by **one** • change the last visit date to the current date • the changed values must be available to the code that follows the call to `UpdateVisit()`

(a) (i) Write pseudocode for module `UpdateVisit()`

Assume the customer has made at least **one** previous visit.

[5]

(ii) The programmer decides to amend the `UpdateVisit()` module so that loyalty points are further increased if the customer visits the coffee shop the same day of the week as their last visit.

Write the pseudocode for this condition.

[1]

(b) A text file `Loyalty.txt` will be used to store the data items for the loyalty scheme.

The text file `Loyalty.txt` contains only one line of data for each customer.

Each data item is separated by a comma:

`<CustomerID>,<LoyaltyPointsString>,<LastVisitDateString>`

`LastVisitDateString` is always eight characters in length in the format: DDMMYYYY.

An example of how a line of data will be stored in the text file `Loyalty.txt` is:

`"100123,132,05032025"`

This example shows that the customer with an ID of 100123 had 132 loyalty points following their last visit to the coffee shop on 05/03/2025.

If a customer visits the coffee shop on a Monday, the value of their loyalty points is increased by 10.

The programmer has defined **two** more program modules:

Module	Description
`FindCustomer()` (is already written)	• called with a parameter of type `STRING` that represents a customer ID • returns the line of data read from the text file `Loyalty.txt` containing the data items for that customer
`MondayCheck()`	• called with a parameter of type `STRING` that represents the customer ID of a customer • calls `FindCustomer()` • extracts the loyalty points from the string returned by `FindCustomer()` • increases the loyalty points for the current customer by 10 if the day visited is a Monday • returns an integer value representing the loyalty points

(i) Write pseudocode for module `MondayCheck()`

The module `FindCustomer()` must be used and assume it returns a valid string for the current customer.

[8]

(ii) A date is stored in `LastVisitDateString` in the format:

DDMMYYYY

DD is a 2-digit string
MM is a 2-digit string
YYYY is a 4-digit string.

For example, the date 03/09/2024 is stored as `"03092024"`

An algorithm is needed to convert the string stored in `LastVisitDateString` to data type `DATE` which is then stored in the variable `LastVisitDate`.

Complete the pseudocode for this algorithm:

Assume that `LastVisitDateString` has been declared and contains valid data.

```
DECLARE DayInt, MonthInt, YearInt : INTEGER

DECLARE LastVisitDate : DATE
```

[4]