INSTRUCTIONS TO CANDIDATES

Answer ALL of questions 1, 2, 3 and 4.
Answer only ONE section of question 5. This is the section which requires you to use the Integrated Development Environment (IDE) of your chosen programming language.
You will need to record all of your answers to questions 1, 2, 3 and 4 in a single word processed document.

INFORMATION FOR CANDIDATES

The number of marks is given in brackets at the end of each question or part-question.
You are reminded of the need for good English and orderly, clear presentation in your answers.
The total number of marks available is 60.
You will need a computer with an installed functional copy of the Integrated Development Environment (IDE) appropriate to your chosen programming language and word processing software.
A calculator is allowed in this examination.
Remember to save your work regularly.
Scenario

StarLight Coffee Shop

StarLight Coffee is a small independent retailer that sells coffee to take away or enjoy in its modern café setting.

The manager of StarLight Coffee has decided to start a loyalty scheme. The scheme will reward customers with loyalty points for purchases made. Customers who collect enough points will be able to trade these for cups of coffee.

The manager has decided to commission a new computerised system to record sales and manage the loyalty scheme. The system will need to store customers’ details, such as customer account number, first name, surname, postcode and number of loyalty points collected.

You have been commissioned to develop the new computer system for StarLight Coffee.
1. The class diagram represents the relationships between some of the classes that StarLight Coffee will need for their customer loyalty scheme.

```
Transaction

#accountNumber : Real
+transactionID : Integer

+getAccountNumber() : Real
+makeNewTransactionID()
+getTransactionID() : Integer
```

```
AddCredit

-amount : Integer
-type : String
-date : Date

+setAmount(Integer)
+setType(String)
+setDate(Date)
+execute(Real)
```

```
Withdrawal

-amount : Integer
-date : Date

+setAmount(Integer)
+setDate(Date)
+execute(Real)
```

```
BalanceEnquiry

-successful : Boolean

+execute(Real): Integer
+success() : Boolean
```
(a) State the name of an identifier for a subclass of Transaction. [1]

(b) The execute method always accepts a parameter.
   (i) State the parameter data type for the method execute. [1]
   (ii) One instance of execute returns a value. State the return data type for this execute method. [1]
   (iii) State within which subclass the method execute returns a value. [1]

(c) State the name of the most appropriate identifier for a variable that is used to store the following:
   (i) TRUE [1]
   (ii) "Coffee" [1]

(d) Explain the significance and implication of the + symbol in the attribute +success() : Boolean in the class BalanceEnquiry. [2]

(e) An object called transaction1 is instantiated from the class AddCredit.
   Applying the concept of inheritance, list all of the methods that would be contained within the object transaction1. [2]

(f) An object called transaction2 is instantiated from the class Withdrawal.
   Applying the concept of inheritance, list all of the attributes that would be accessible within the object transaction2. [2]

(g) Explain why the attribute accountNumber is situated within the class Transaction. [2]

Save your work for Question 1 as a word processed document named Answers.
2. **StarLight Coffee** will need a backup policy and will need to consider different methods of backing up the customer data.

Explain the purpose of the backup policy and describe different methods **StarLight Coffee** could use for backing up its data. Your answer should include different secondary storage devices **StarLight Coffee** could use. [6]

Save your work for Question 2 in the word processed document named Answers.
3. **StarLight Coffee** have decided to incentivise the purchase of certain products by offering differing numbers of points for different products.

**StarLight Coffee** uses an array called `productPoints[]` to lookup the number of points for different products sold. Each product ID has a number of points per whole pound spent assigned to it. The algorithm below calculates and outputs data on the purchases.

```plaintext
1  productPoints[10,2] is real {array to use}
2  productID is integer
3  amountSpent is integer
4  numberOfPurchases is integer
5  subTotal is real
6  cTotal is real
7  set subTotal = 0
8  set cTotal = 0
9
10  for i = 0 to 9
11      productPoints[i,0] = i + 1
12      productPoints[i,1] = (i + 1) * 1.1
13  next i
14
15  output "Enter the number of purchases in this sale:"
16  input numberOfPurchases
17
18  for j = 1 to numberOfPurchases
19      output "Enter product ID"
20      input productID
21
22      output "Enter amount spent (£)"
23      input amountSpent
24
25      output "Points per product:"
26      output productPoints[productID, 1]
27
28      subTotal = productPoints[productID, 1] * amountSpent
29
30      cTotal = cTotal + subTotal
31
32      output "Cumulative total:" + cTotal
33
34  next j
35
36  End
```

Copy and complete the table below showing the outputs of the algorithm for the inputs provided. **When numberOfPurchases = 4**

<table>
<thead>
<tr>
<th>productID</th>
<th>Amount spent (£)</th>
<th>Points per product</th>
<th>Cumulative total</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Save your work for Question 3 in the word processed document named Answers.
4. **StarLight Coffee** has decided to award a ‘pm bonus’ to customers who purchase coffees during the afternoon.

Using a recognised convention, design an algorithm to help **StarLight Coffee** calculate ‘pm bonus’ awards.

The algorithm must allow a user to input a time in 24-hour format. The algorithm should:

- repeatedly accept an integer input called `currentFullTime`
- convert the time entered into the 12-hour format, followed by `am` or `pm`
- output if a bonus has been awarded
- if applicable, set a Boolean flag called `bonusAwarded` to `True`

For example:

If the user enters 0013, the program outputs: “12:13 AM No Bonus”
If the user enters 1834, the program outputs: “6:34 PM Bonus Awarded”

Your algorithm should be written in pseudo-code, using self-documenting identifiers.  

Save your work for Question 4 in the word processed document named Answers.
5. Select the programming language of your choice from section (a), (b) or (c) and answer all questions in your chosen section.

(a) Visual Basic

StarLight Coffee wants a computer system to be developed using Visual Basic that meets the requirements outlined below:

- To store customer details
- To recall and count customers with specific details
- To calculate integer additions and subtractions in a simple calculator
- To store and recall calculations carried out on the calculator

(i) Open the file coffeecustomer
   - Read through the code and familiarise yourself with its contents
   - The file contains incomplete code, which is intended to allow StarLight Coffee to store and count customers with specific details

   Complete this code. \[4\]

   Remember to save the changes made to the file coffeecustomer

(ii) Create a new program that will allow StarLight Coffee to:

   - Input whole integer numbers 1 to 9
   - Carry out the addition (+) operator
   - Carry out the subtraction (-) operator
   - Display the result of either addition or subtraction on screen
   - Store the last calculation result on disk in a text file called calcmemory
   - Confirm storage of the result on screen
   - Retrieve the last calculation result from disk \[12\]

   Save your work as coffeecalculator

(iii) Using the internal facility of Visual Basic, add annotated listings to your code from question 5(a)(ii) that would clearly explain the design of your program to another software developer. \[4\]

   Save your annotations in the same file as 5(a)(ii) above.
(b) **Java**

**StarLight Coffee** wants a computer system to be developed using **Java** that meets the requirements outlined below:

- To store customer details
- To recall and count customers with specific details
- To calculate integer additions and subtractions in a simple calculator
- To store and recall calculations carried out on the calculator

(i) Open the file `coffeecustomer`
- Read through the code and familiarise yourself with its contents
- The file contains incomplete code, which is intended to allow **StarLight Coffee** to store and count customers with specific details

Complete this code. **[4]**

**Remember to save the changes made to the file `coffeecustomers`**

(ii) Create a new form that will allow **StarLight Coffee** to:

- Input whole integer numbers 1 to 9
- Carry out the addition (+) operator
- Carry out the subtraction (-) operator
- Display the result of either addition or subtraction on screen
- Store the last calculation result on disk in a text file called `calcmemory`
- Confirm storage of the result on screen
- Retrieve the last calculation result from disk **[12]**

**Save your work as `coffeecalculator`**

(iii) Using the internal facility of **Java**, add annotated listings to your code from question 5(b)(ii) that would clearly explain the design of your program to another software developer. **[4]**

**Save your annotations in the same file as 5(b)(ii) above.**
(c) **Python**

**StarLight Coffee** wants a computer system to be developed using **Python** that meets the requirements outlined below:

- To store customer details
- To recall and count customers with specific details
- To calculate integer additions and subtractions in a simple calculator
- To store and recall calculations carried out on the calculator

(i) Open the file `coffeecustomer`

- Read through the code and familiarise yourself with its contents
- The file contains incomplete code, which is intended to allow **StarLight Coffee** to store and count customers with specific details

Complete this code.

**Remember to save the changes made to the file coffeecustomer**

(ii) Create a new form that will allow **StarLight Coffee** to:

- Input whole integer numbers 1 to 9
- Carry out the addition (+) operator
- Carry out the subtraction (-) operator
- Display the result of either addition or subtraction on screen
- Store the last calculation result on disk in a text file called `calcmemory`
- Confirm storage of the result on screen
- Retrieve the last calculation result from disk

**Save your work as coffeecalculator**

(iii) Using the internal facility of **Python**, add annotated listings to your code from question 5(c)(ii) that would clearly explain the design of your program to another software developer.

**Save your annotations in the same file as 5(c)(ii) above.**

**END OF PAPER**