User Defined Data types

Question 1

2 Data types can be classified as composite or non-composite.

A record is declared of type `box` using the following pseudocode.

```
TYPE size = (small, medium, large)

TYPE box

    DECLARE volume : size

    DECLARE price : REAL

    DECLARE colour : STRING

ENDTYPE

DECLARE myBox : ARRAY [1:6] OF box
```

(a) (i) Identify **one** composite and **three** non-composite data types used in the pseudocode.

Composite data type ....

Non-composite data type 1 ....

Non-composite data type 2 ....

Non-composite data type 3 ....
[4]

(b) A box is red, with medium volume and a price of $10.99.

Write **pseudocode** to store the details of this box in the first element of the array.

....

....

....

....

.... [3]

Question 2

2 Data types can be classified as composite or non-composite.

(a) Draw **one** line from each data type to its correct classification.

Data type	Classification
Pointer	
Record	Composite
Set	
Class	Non-composite
Integer	

[2]

(b) A user-defined data type, `timeOfDay`, is declared using the following pseudocode.

```
TYPE timeOfDay = (morning, afternoon, evening, night)
```

(i) Identify the type of user-defined data type declared **and** state its classification.

Type ....

Classification ....
[2]

(ii) Write pseudocode to declare the variable `session` of type `timeOfDay`.
Assign the value `afternoon` to the variable `session`.

....

....

....

.... [2]

Question 3

2   Data types can be classified as composite or non-composite.

A record is declared of type `box` using the following pseudocode.

```
TYPE size = (small, medium, large)

TYPE box

    DECLARE volume : size

    DECLARE price : REAL

    DECLARE colour : STRING

ENDTYPE

DECLARE myBox : ARRAY [1:6] OF box
```

(a) (i) Identify **one** composite and **three** non-composite data types used in the pseudocode.

Composite data type ....

Non-composite data type 1 ....

Non-composite data type 2 ....

Non-composite data type 3 ....

[4]

(ii) Identify the data type in the pseudocode that is enumerated.

.... [1]

(b) A box is red, with medium volume and a price of $10.99.

Write **pseudocode** to store the details of this box in the first element of the array.

....

....

....

....

.... [3]

Question 4

5 **(a)** Explain why user-defined data types are necessary.

....

....

....

.... [2]

(b) An organisation stores data about its employees.

- Employee ID is a five-digit number, for example, 01234.
- Employee name is a string, for example, 'Kiri Moana'.
- Department is one of three values: Sales, Technical, Customer services.
- Salary is an integer value in the range 25 000 to 150 000.

(i) Complete the following **pseudocode** definition of a user-defined data type to store the employee data.

```
TYPE Employee

   DECLARE EmployeeID   :  ...................................................................

   DECLARE EmployeeName :  STRING

   DECLARE Department   :  ( ................................................................

                           ....................................................................)

   DECLARE Salary       :  25000..150000
```

....
[4]

(ii) Write a **pseudocode** statement to declare a variable, NewEmployee of data type Employee.

....

.... [1]

(iii) Write a **pseudocode** statement that assigns 02244 to the EmployeeID of NewEmployee.

....

.... [1]

(iv) Employee is an example of a record that is a composite data type.

State **two** other composite data types.

1 ....

2 ....
[2]

Question 5

i

6 (a) State what is meant by a **user-defined data type**.

....

.... [2]

(b) A pseudocode declaration for a user-defined data type for the months of the year is as follows:

```
TYPE
   DECLARE Months: (January, February, March, April, May, June, July,
                    August, September, October, November, December)
ENDTYPE
```

(i) Identify this type of user-defined data type.

....

.... [1]

(ii) Write a **pseudocode** statement to declare a variable `CurrentMonth` of data type `Months`.

....

.... [1]

(iii) Write a **pseudocode** statement to assign the value `August` to the variable `CurrentMonth`.

....

.... [1]

Question 6

2 A programmer uses non-composite and composite data types to create a program.

(a) Define the term **non-composite data type**.

....

....[1]

(b) Describe **two** different non-composite data types.

Data type 1 ....

Description ....

....

....

Data type 2 ....

Description ....

....

....

[4]

(c) Define the term **composite data type**.

....

....[1]

(d) Describe **two** different composite data types.

Data type 1 ....

Description ....

....

....

Data type 2 ....

Description ....

....

....

[4]

Question 7

1   Data types can be defined in a programming language.

The data type, StudentRecord, is defined by the code:

```
TYPE StudentRecord
    DECLARE StudentID        : INTEGER
    DECLARE StudentFirstName : STRING
    DECLARE StudentSurname   : STRING
    DECLARE StudentDOB       : DATE
    DECLARE StudentCourse    : ARRAY[1:10] OF STRING
ENDTYPE
```

A variable, CollegeStudent, is declared with the code:

```
DECLARE CollegeStudent : StudentRecord
```

(a)   Write a pseudocode statement to assign 6539 to the StudentID of CollegeStudent.

....[1]

(b)   The type definition for StudentRecord is changed.

(i)   Students can take six courses from: Computer Science, Engineering, Science, Maths, Physics, Chemistry, Music, Drama and English Language.

Rewrite **one** line from the type definition of StudentRecord to implement the change.

DECLARE ....

....

....

....[2]

(ii)   The values for the field StudentID must be between 1 and 8000 inclusive.

Rewrite **one** line from the type definition of StudentRecord to implement the change.

DECLARE ....[1]

(c) A programmer is asked to write a program to process the assessment data for each student. Students sit one exam in every course they take.

A composite data type, StudentAssessment, needs to be defined with the following three fields.

- a student assessment code (a unique code of three letters and two digits)
- the marks for the six exams
- the average mark of the six exams

(i) Write **pseudocode** to define the data type StudentAssessment.

....

....

....

....

....

....[4]

(ii) Data about all students and their assessments are stored in a file that uses random organisation. The StudentID is used as the key field.

The program allows a user to enter data for a new student.

Explain how the program adds the new data to the file.

....

....

....

....

....

....[3]

Question 8

1   Consider the following user-defined data type.

```
TYPE Book
    DECLARE ISBN     : INTEGER
    DECLARE Author   : STRING
    DECLARE Title    : STRING
    DECLARE Supplier : (Amazone, Stones, Smiths, Blackwalls, Greens,
                        Coals, Boarders)
ENDTYPE
```

(a) Name the data type of `Book`.

....[1]

(b) Name the non-composite data type used in the `Supplier` declaration.

....[1]

(c) (i) Write a pseudocode statement to declare a variable, `BestSeller`, of type `Book`.

....[1]

    (ii) Write a pseudocode statement to assign "`John Williams`" to the author of
          `BestSeller`.

....[1]

Question 9

1   (a) Consider the following user-defined data type:

```
TYPE LibraryBookRecord
    DECLARE ISBN     : INTEGER
    DECLARE Title    : STRING
ENDTYPE
```

(i) Write a pseudocode statement to declare a variable, `Book`, of type `LibraryBookRecord`.

....[1]

(ii) Write a pseudocode statement that assigns 'Dune' to the `Title` of `Book`.

....[1]

(b) The user-defined data type `LibraryBookRecord` needs to be modified by adding the following fields:

- a field called `Genre` which can take two values, fiction or non-fiction
- a field called `NumberOfLoans` which can be an integer value in the range 1 to 99

Write the updated version of `LibraryBookRecord`.

....

....

....

....

....

....

....**[3]**

(c) A pointer is a variable that stores the address of a variable of a particular type.

Consider the code on page 3, which uses the following identifiers:

Identifier	Data type	Description
`IntPointer`	`^INTEGER`	pointer to an integer
`IntVar`	`INTEGER`	an integer variable
`Temp1`	`INTEGER`	an integer variable
`Temp2`	`INTEGER`	an integer variable

```
IntVar ← 57              // assigns the value 57 to the integer
                         // variable IntVar
IntPointer ← @IntVar     // assigns to IntPointer the address of the
                         // integer variable IntVar
Temp2 ← IntPointer^      // assigns to variable Temp2 the value at an
                         // address pointed at by IntPointer
IntPointer^ ← Temp1      // assigns the value in the variable Temp1 to
                         // the memory location pointed at by IntPointer
```

The four assignment statements are executed. The diagram shows the memory contents after execution.

Variable	Memory address	Contents
	. . .	
	8217	
IntVar	8216	88
	8215	
	8214	
	. . .	
	7307	
IntPointer	7306	8216
	7305	
	. . .	
	6717	
Temp1	6716	88
Temp2	6715	57
	6714	
	. . .	

Use the diagram to state the current values of the following expressions:

(i) `@Temp2` ....[1]

(ii) `IntPointer` ....[1]

(iii) `IntPointer^` ....[1]

(iv) `IntPointer^ = Temp2 + 6` ....[1]

(d) Write pseudocode statements that will achieve the following:

(i) Assign the value `22` to the variable `Temp2`.

....[1]

(ii) Place the address of `Temp1` in `IntPointer`.

....[1]

(iii) Copy the value in `Temp2` into the memory location currently pointed at by `IntPointer`.

....[1]

Question 10

1 **(a)** Consider the following pseudocode user-defined data type:

```
TYPE MyContactDetail
    DECLARE Name        : STRING
    DECLARE HouseNumber : INTEGER
ENDTYPE
```

(i) Write a pseudocode statement to declare a variable, `NewFriend`, of type `MyContactDetail`.

....[1]

(ii) Write a pseudocode statement that assigns 129 to the `HouseNumber` of `NewFriend`.

....[1]

(b) The user-defined data type `MyContactDetail` needs to be modified by:

• adding a field called `Area` which can take three values, uptown, downtown or midtown
• amending the field `HouseNumber` so that house numbers can only be in the range 1 to 499.

Write the updated version of `MyContactDetail`.

....

....

....

....

....

....

....[3]

(c) A pointer is a variable that stores the address of a variable of a particular type.

Consider the pseudocode on page 3, which uses the following identifiers:

Identifier	Data type	Description
IPointer	^INTEGER	pointer to an integer
Sum	INTEGER	an integer variable
MyInt1	INTEGER	an integer variable
MyInt2	INTEGER	an integer variable

```
Sum ← 91            // assigns the value 91 to the integer variable Sum
IPointer ← @Sum     // assigns to IPointer the address of the
                    // integer variable Sum
MyInt1 ← IPointer^  // assigns to variable MyInt1 the value at an
                    // address pointed at by IPointer
IPointer^ ← MyInt2  // assigns the value in the variable MyInt2 to
                    // the memory location pointed at by IPointer
```

The four assignment statements are executed. The diagram shows the memory contents after execution.

Variable	Memory Address	Contents
	. . .	
	5848	
	5847	
IPointer	5846	4402
	5845	
	. . .	
	4403	
Sum	4402	33
	4401	
	. . .	
	3428	
MyInt1	3427	91
MyInt2	3426	33
	3425	
	. . .	

Use the diagram to state the current values of the following expressions:

(i) IPointer ....[1]

(ii) IPointer^ ....[1]

(iii) @MyInt1 ....[1]

(iv) IPointer^ = MyInt2 ....[1]

(d) Write pseudocode statements that will achieve the following:

    (i) Place the address of `MyInt2` in `IPointer`.

        ....[1]

    (ii) Assign the value `33` to the variable `MyInt1`.

        ....[1]

    (iii) Copy the value in `MyInt2` into the memory location currently pointed at by `IPointer`.

        ....[1]

Question 11

3   **(a)** A particular programming language allows the programmer to define their own data types.

       `ThisDate` is an example of a user-defined structured data type.

```
TYPE ThisDate
    DECLARE ThisDay     :  (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,
                           13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23,
                           24, 25, 26, 27, 28, 29, 30, 31)
        DECLARE ThisMonth   :  (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug,
                           Sep, Oct, Nov, Dec)
        DECLARE ThisYear    :  INTEGER
ENDTYPE
```

A variable of this new type is declared as follows:

```
DECLARE DateOfBirth : ThisDate
```

    (i) Name the non-composite data type used in the `ThisDay` and `ThisMonth` declarations.

        ....[1]

    (ii) Name the data type of `ThisDate`.

        ....[1]

    (iii) The month value of `DateOfBirth` needs to be assigned to the variable `MyMonthOfBirth`.

       Write the required statement.

        ....[1]

(b) Annual rainfall data from a number of locations are to be processed in a program.

The following data are to be stored:

- location name

- height above sea level (to the nearest metre)

- total rainfall for each month of the year (centimetres to 1 decimal place)

A user-defined, composite data type is needed. The programmer chooses `LocationRainfall` as the name of this data type.

A variable of this type can be used to store all the data for one particular location.

(i) Write the definition for the data type `LocationRainfall`.

....

....

....

....

....

....

....[5]

Question 12

4 **(a)** A particular programming language allows the programmer to define their own data types.

An example of a user-defined data type for an address is:

```
TYPE ThisAddress
    DECLARE ThisHouseNo : INTEGER
    DECLARE ThisStreet  : STRING
    DECLARE ThisTown    : STRING
ENDTYPE
```

A variable of this new type is declared as follows:

```
DECLARE HomeAddress : ThisAddress
```

(i) Write the statement that assigns the house number 34 to `HomeAddress`.

....[1]

(ii) The type definition for `ThisAddress` is to be changed.

Rewrite one line from the definition for each of the following changes.

House numbers are in the range from 1 to 10.

DECLARE ....

The possible towns are limited to: Brightown, Arunde and Shoram.

DECLARE ....[2]

(b) Temperature data from a number of weather stations are to be processed by a program.

The following data are to be stored:

- weather station ID (a unique four-letter code)

- latitude (to 2 decimal places)

- average temperature (to the nearest whole number) for each year from 2001 to 2015 inclusive

A programmer designs a composite data type `WeatherStation`. A variable of this type can be used to store all the data for one particular station.

(i) Write the definition for the user-defined data type `WeatherStation`.

....

....

....

....

....

....[5]

Answer

Answer 1

2(a)(i)	Composite
	   `box`
	Non-composite
	   `size` / enumerated
	   `REAL`
	   `STRING`
		4

2(a)(ii)	`size`	**1**

2(b)	`myBox[1].volume ← medium`
	`myBox[1].price ← 10.99`
	`myBox[1].colour ← "red"`
		3

Answer 2

2(a)	**2** marks for all 5 single lines correct **1** mark for 4 lines correct otherwise zero **Data type** **Classification** Pointer Record ——— Composite Set Class Integer ——— Non-composite	**2**
2(b)(i)	Type **Enumerated** Classification **Non-composite**	**2**
2(b)(ii)	`DECLARE session : timeOfDay` `session ← afternoon`	**2**

Answer 3

2(a)(i)	Composite
	   `box`
	Non-composite
	   `size` / enumerated
	   `REAL`
	   `STRING`
		4

2(a)(ii)	`size`	**1**

2(b)	`myBox[1].volume ← medium`
	`myBox[1].price ← 10.99`
	`myBox[1].colour ← "red"`
		3

Answer 4

5(a)	**1 mark** per bullet point to **max 2** • No suitable data type is provided by the language used • The programmer needs specify a new data type • … that meets the requirements of the application / program	2

5(b)(i)	**1 mark** per bullet point
	
	• `EmployeeID` declared as `STRING`
	• `Sales, Technical` and `CustomerServices` ...
	• ... with commas in-between
	• `ENDTYPE`
	
	<pre>TYPE Employee
	    DECLARE EmployeeID : STRING
	    DECLARE EmployeeName : STRING
	    DECLARE Department : (Sales, Technical,
	                            CustomerServices)
	    DECLARE Salary : 25000..150000
	ENDTYPE</pre>
		4

5(b)(ii)	`DECLARE NewEmployee : Employee`	1
5(b)(iii)	`NewEmployee.EmployeeID ← "02244"`	1
5(b)(iv)	**1 mark** per bullet point to **max 2** • Array • List • Set • Collection • Class • Stack • Queue • Linked list • Dictionary	2

Answer 5

6(a)	1 mark per bullet point to **max 2** • Derived from one or more existing data types • Used to extend the built-in data types • Creates data-types specific to applications // programmer's requirements	2
6(b)(i)	Enumerated (data type)	1
6(b)(ii)	`DECLARE CurrentMonth : Months`	1
6(b)(iii)	`CurrentMonth ← August`	1

Answer 6

2(a)	<u>single data type</u> that does not involve a reference to another type/usually built in to a programming language	1
2(b)	**1 mark for data type, 1 for definition, max 4, 2 data types** ∞ Integer ∞ Stores a whole number ∞ Boolean ∞ Stores true or false/1 or 0/on or off ∞ Real/Single/Double/Float/Decimal ∞ Stores decimal numbers ∞ String ∞ Stores zero or more characters ∞ Char ∞ Stores a single character ∞ Pointer ∞ Whole number used to reference a memory location	4
2(c)	data type constructed from other data types	1
2(d)	**1 mark for naming, 1 for description, max 4, 2 data types** ∞ Record ∞ collection of related items which may have different data types ∞ Array ∞ (Indexed) collection of items with the same data type ∞ List ∞ (Indexed) collection of items that can have different data types ∞ Set ∞ stores a finite number of different values that have no order // supports mathematical operations ∞ Class/Structure ∞ Gives the properties and methods for an object	4

Answer 7

1(a)	`CollegeStudent.StudentID ← 6539`	1
1(b)(i)	**1 mark per bullet** • `StudentCourse: ARRAY[1:6] OF` • All valid string options , for example: `DECLARE StudentCourse: ARRAY[1:6] OF ("Computer Science", "Engineering", "Science", "Maths", "Physics", "Chemistry", "Music", "Drama", "English Language")`	2
1(b)(ii)	`DECLARE StudentID: 1 .. 8000`	1

1(c)(i)	**1 mark per bullet**
	
	• Type declaration `TYPE` and `ENDTYPE`
	• Declaring `Code` as `STRING`
	• Declaring `Mark` as `ARRAY [1:6] OF INTEGER`
	• `AverageMark` as `REAL`
	
	For example:
	`TYPE StudentAssessment`
	  `DECLARE Code        : STRING`
	  `DECLARE Mark        : ARRAY[1:6] OF INTEGER`
	  `DECLARE AverageMark : REAL`
	`ENDTYPE`
		4

1(c)(ii)	Any **3** from, **1** mark per bullet	**3**
	• `StudentID`/key field is hashed to produce home location • If home location is free, insert record/data • Else use overflow method to find free location to store record / data • If no free location available then file is full and record/data cannot be stored	

Answer 8

1	(a) Record	**1**
	(b) Enumerated	**1**
	(c) `DECLARE BestSeller : Book`	**1**
	(d) `BestSeller.Author ← "John Williams"`	**1**

Answer 9

1(a)(i)	`DECLARE Book : LibraryBookRecord`	**1**
1(a)(ii)	`Book.Title ← "Dune"`	**1**

1(b)	`TYPE LibraryBookRecord`
	`    DECLARE ISBN        : INTEGER`
	`    DECLARE Title       : STRING`
	`    DECLARE Genre       : (Fiction, Non-Fiction)` **1**
	`    DECLARE NumberOfLoans : 1 .. 99` **1**
	`ENDTYPE`
	mark for correct declaration and first two fields (**note**: only if attempt at modification) **1**
		3

1(c)(i)	6715	**1**
1(c)(ii)	8216	**1**
1(c)(iii)	88	**1**
1(c)(iv)	`FALSE`	**1**
1(d)(i)	`Temp2 ← 22`	**1**
1(d)(ii)	`IntPointer ← @Temp1`	**1**
1(d)(iii)	`IntPointer^ ← Temp2`	**1**

Answer 10

1(a)(i)	`DECLARE NewFriend  : MyContactDetail`		**1**
1(a)(ii)	`NewFriend.HouseNumber ← 129`		**1**

1(b)	Declaration of `Name, Area, HouseNumber`
	 Inclusion of three correct values for `Area`
	 Inclusion of correct range for `HouseNumber`
	
	 For example:
	
	 `TYPE MyContactDetail`
	 `    DECLARE Name        : STRING`
	 `    DECLARE Area        : (uptown, downtown, midtown)`
	 `    DECLARE HouseNumber : 1..499`
	 `ENDTYPE`
		1
		 1
		 1
		
		 1
		 1 & 1
			3

1(c)(i)	4402		**1**
1(c)(ii)	33		**1**
1(c)(iii)	3427		**1**
1(c)(iv)	`TRUE`		**1**
1(d)(i)	`IPointer ← @MyInt2`		**1**
1(d)(ii)	`MyInt1 ← 33`		**1**
1(d)(iii)	`IPointer^ ← MyInt2`		**1**

Answer 11

3 (a) (i)	enumerated	1
(ii)	record	1
(iii)	`MyMonthOfBirth ← DateOfBirth.ThisMonth`	1

(b) (i)	`TYPE LocationRainfall`
	 `    DECLARE LocationName        : STRING`
	 `    DECLARE LocationHeight      : INTEGER`
	 `    DECLARE TotalMonthlyRainfall : ARRAY[1..12] OF REAL`
	 `ENDTYPE`
		1
		 1
		 1
		 1 +1

Answer 12

4 (a) (i)	`HomeAddress.ThisHouseNo ← 34`	1
(ii)	`DECLARE ThisHouseNo: 1..10`	1
	`DECLARE ThisTown: [Brightown, Arunde, Shoram]`	1

(b) (i)	`TYPE WeatherStation`
	 `    DECLARE StationID   : STRING`
	 `    DECLARE Latitude    : REAL`
	 `    DECLARE Temperature : ARRAY[1..15] OF INTEGER`
	 `ENDTYPE`
		1
		 1
		 1 + 1
		 1