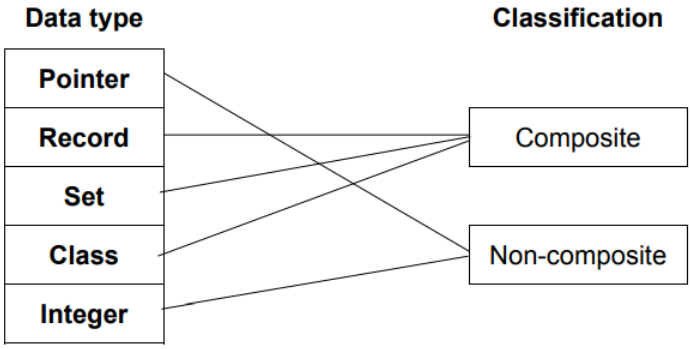
Answer

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 	**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`
	
	`TYPE Employee`
	`  DECLARE EmployeeID  : STRING`
	`  DECLARE EmployeeName : STRING`
	`  DECLARE Department  : (Sales, Technical,`
	`  CustomerServices)`
	`  DECLARE Salary  : 25000..150000`
	`ENDTYPE`
		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)	single data type 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 • `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	**3**

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` 1
	Inclusion of three correct values for `Area` 1
	Inclusion of correct range for `HouseNumber` 1
	
	For example:
	
	`TYPE MyContactDetail`
	`  DECLARE Name : STRING`
	`  DECLARE Area : (uptown, downtown, midtown)` **1**
	`  DECLARE HouseNumber : 1..499` **1 & 1**
	`ENDTYPE`
		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`
	
	`DECLARE ThisTown: [Brightown, Arunde, Shoram]`
		1
		
		1

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