# QUESTION 1.

**3** **(a)** A particular programming language allows the programmer to define their ow

`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 prog

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]

**(ii)** The programmer decides to store all the data in a file. Initially, data from 27 locations will be stored. More rainfall locations will be added over time and will never exceed 100.

The programmer has to choose between two types of file organisation. The two types are serial and sequential.

Give **two** reasons for choosing serial file organisation.

.....

.....

.....

.....[2]

# QUESTION 2.

**4** **(a)** A particular programming language allows the programmer to define their ow

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

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]

**(ii)** The programmer decides to store all the data in a file. The number of weather stations could grow to reach 20000, but not all stations will be present at first.

The programmer decides on random organisation for the file.

Describe **three** steps which show how a new weather station record is added to the file.

1 .....

.....

2 .....

.....

3 .....

.....[3]

# QUESTION 3.

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 intege
                        // variable IntVar
IntPointer ← @IntVar    // assigns to IntPointer the address of
                        // 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`.

.....

**(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 4.

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
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`.

.....

**(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 5.

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 6.

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

.....

.....

**(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]