

1 A program stores data about birds using Object-Oriented Programming (OOP).

The class `Bird` stores the data about the birds:

Bird	
<code>DistancePerHour : Real</code>	stores the number of kilometres per hour (km/h) the bird can fly (between 0.0 and 100.0 inclusive)
<code>Species : String</code>	stores the species of the bird, for example Pigeon
<code>XPosition : Real</code>	stores the current horizontal position of the bird
<code>YPosition : Real</code>	stores the current vertical position of the bird
<code>Constructor ()</code>	initialises <code>Species</code> and <code>DistancePerHour</code> to the parameter values; initialises <code>XPosition</code> and <code>YPosition</code> to 500.0
<code>GetPosition ()</code>	returns a string message that contains the horizontal and vertical position of the bird
<code>GetSpecies ()</code>	returns the species of the bird
<code>Move ()</code>	takes a direction and number of minutes flying (between 0 and 500 inclusive) as parameters and updates the appropriate horizontal or vertical position

(a) (i) Write program code to declare the class `Bird` and its constructor.

Do **not** declare the other methods.

All attributes should be private.

Use your programming language appropriate constructor.

If you are writing in Python, include attribute declarations using comments.

Save your program as **Question1_N25**.

Copy and paste the program code into part **1(a)(i)** in the evidence document.

[4]

(ii) The method `GetSpecies ()` returns the species of the bird.

Write program code for `GetSpecies ()`

Save your program.

Copy and paste the program code into part **1(a)(ii)** in the evidence document.

[2]

(iii) The method `GetPosition ()` returns a string of the bird's data in the format:

"X = " & <XPosition> & " Y = " & <YPosition>

An example string for a bird is:

X = 500.0 Y = 250.0

Write program code for `GetPosition ()`

Save your program.

Copy and paste the program code into part **1(a)(iii)** in the evidence document.

[3]

(iv) If the bird is travelling north, the vertical position increases. If the bird is travelling south, the vertical position decreases.

If the bird is travelling east, the horizontal position increases. If the bird is travelling west, the horizontal position decreases.

The method `Move ()`:

- takes the direction of travel as a parameter in the form 'N' for north, 'S' for south, 'E' for east or 'W' for west
- takes the number of minutes that the bird has been flying (to the nearest minute) as a parameter
- calculates the distance travelled by the bird using the formula:
distance travelled = (distance per hour/60) * minutes flying
- updates the vertical or horizontal position using the distance calculated.

You do **not** need to validate the parameters.

Write program code for `Move ()`

Save your program.

Copy and paste the program code into part **1(a)(iv)** in the evidence document.

[5]

(b) The main program creates two instances of `Bird`

The first bird is the species 'Cockatiel' and flies 71.0 km/h.

The second bird is the species 'Macaw' and flies 56.0 km/h.

Write program code to declare and initialise the **two** birds.

Save your program.

Copy and paste the program code into part **1(b)** in the evidence document.

[3]

(c) (i) The main program needs to:

- output a message that includes the species and current X position and Y position for each bird
- prompt the user to select one of the birds to move and take this as an input
- prompt the user to enter the direction the bird has been travelling and take this as an input
- prompt the user to enter the time to the nearest minute that the bird has been travelling and take this as an input
- call the appropriate method to update the chosen bird's position
- output an update on the bird's new position.

Each input needs to repeat until valid data is entered. All outputs must be meaningful.

Write program code to amend the main program.

Save your program.

Copy and paste the program code into part **1(c)(i)** in the evidence document.

[8]

(ii) Test your program **four** times to meet these criteria:

Test 1: The Cockatiel travels north for 60 minutes.

Test 2: The Macaw travels south for 30 minutes.

Test 3: The Cockatiel travels west for 30 minutes.

Test 4: The Macaw travels east for 60 minutes.

Take a screenshot of the output(s).

Save your program.

Copy and paste the screenshot(s) into part **1(c)(ii)** in the evidence document.

[3]