⚠ Ant Simulation ⚠

C# Theory Questions

These questions refer to the **Preliminary Material** and the **Skeleton Program**,
but **do not** require any additional programming.

TOTAL MARKS: 54

1. The **Main** procedure in the pre-release program asks the user to enter several pieces of data. One of those is **the simulation number**.

   Currently, the program assumes the user will always enter valid numeric input inside the range 1–4 inclusive.

   (a) Explain why this program is not robust when processing user input. [2]

   ....
   ....
   ....

   (b) The code contains a prompt on line 20: **Enter simulation number:**

   What error would occur during runtime if the user typed "four" when receiving the prompt from line 20? [1]

   ....
   ....

   (c) How would a programmer resolve this issue? [1]

   ....
   ....

2. Give an identifier used for:

   (a) A local variable used within the **GetCellReference** procedure. [1]

   ....
   ....

   (b) A class variable. [1]

   ....
   ....

   (c) The loop counter variable used when nests are being created within the constructor of the **Simulation** class. [1]

   ....
   ....

QUESTION 2 CONTINUES OVERLEAF

(d) A variable that is used to store the neighbouring cell indices as a list. [1]

....

....

3. This program makes use of **Encapsulation**.

(a) Explain what encapsulation means in object-oriented programming. [2]

....

....

....

(b) Describe how this concept is used within the **Cell** class. [2]

....

....

....

4. This program makes use of **Inheritance**.

(a) Explain what inheritance means in object-oriented programming. [2]

....

....

....

(b) Give an advantage of using this technique. [1]

....

....

(c) Explain how inheritance is used between the Entity, Ant, and WorkerAnt classes. [2]

....

....

....

5. What data structure is used to represent the grid of cells, and how many elements does it contain in Simulation 3? [2]

....

....

....

6. Describe how the **AdvanceStage** methods demonstrate polymorphism. [2]

....

....

....

7. Explain how the simulation demonstrates the use of abstraction by modelling, and state one advantage of using this technique to simulate the behaviour of ant colonies. [3]

....

....

....

....

....

8. The simulation stores all cells in a one-dimensional list/array.

(a) Explain how the private method **GetIndex** allows the program to work with two-dimensional grid coordinates even though a one-dimensional list is used. [2]

....

....

....

(b) Which index would be returned from a call to this method with the **Row** = 2 and **Column** = 5 if **NumberOfColumns** was 5? [1]

....

....

9. This question relates to the **GetIndicesOfNeighbours** method of the **Simulation** class.

(a) This is a private method. Explain what it means that this method is private. [1]

....

....

(b) What would this method return if it were called with Row = 1 and Column = 1 for a 5×5 simulation grid? [2]

....

....

....

10. Describe the movement choice for an ant who is:

(a) A worker ant that is away from their nest and is carrying food. [1]

....

....

(b) A worker ant that does not detect any pheromone trails nearby and has no food. [1]

....

....

QUESTION 10 CONTINUES OVERLEAF

(c) A worker ant that is not carrying food and does detect pheromone trails nearby. [1]

....

....

(d) A queen ant in her nest cell. [1]

....

....

11. This question is about the public method **AdvanceStage** inside the **Nest** class.

(a) For a nest that currently houses 1 queen and 5 workers, how much food will be consumed by all the ants belonging to this nest when this method is called? [1]

....

....

(b) How does the program make sure that a nest cannot have its food level decremented below 0? [1]

....

....

(c) How many ants would be culled by a call to this method for a nest with a **FoodLevel** of 7 and 10 living ants belonging to it? [1]

....

....

(d) How many ants would be culled by a call to this method for a nest with a **FoodLevel** of 70 and 10 living ants belonging to it? [1]

....

....

(e) Describe the process by which ants are created by a call to this method for a healthy nest with plentiful food for all the ants. [3]

....

....

....

....

....

(f) When the program removes an ant in this method using **Ants.RemoveAt(RPos);** where **RPos** is a random index, what is the time complexity of this operation in Big-O notation, and why? [2]

....

....

....

12. This question is about the **ChooseCellToMoveTo** method in the **Ant** class. This is an abstract method. Describe what this means and how it is used by its subclasses. [2]

....

....

....

13. The program uses random numbers when deciding ant movement and food placement. Explain why these numbers are not completely random and what determines the sequence generated. [2]

....

....

....

14. The **Pheromones** in the program are currently stored in a list in the **Simulation** class. A programmer suggests using a dictionary instead, where the key is the cell coordinates as a tuple and the value is a list of pheromones in that cell.

(a) State one advantage of using a dictionary for this purpose. [1]

....

....

(b) Explain why the coordinate pair used as a dictionary key must be hashable. [2]

....

....

....

15. This program has been written using the object-oriented programming (OOP) paradigm.

Explain two differences between the object-oriented and procedural programming paradigms.

In your answer, you should refer to examples from this program to support your explanation. [4]

....

....

....

....

....

....

END OF QUESTIONS