Theory Questions

These questions are designed to test your understanding of the skeleton code. Many of these are similar to the kinds of question you can expect to see in Section C of the Paper 1 exam. However, sub-questions that are more than 2 marks are rarely seen in this section – these more involved questions are here to challenge your understanding of the code.

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

TOTAL MARKS: 75

1 This question refers to the main subroutine and the constructor of the Puzzle class.

(a) Describe the purpose of the selection statement below, including the meaning/purpose of any parameters for the Puzzle constructor:

```
if len(Filename) > 0:
    MyPuzzle = Puzzle(Filename + ".txt")
else:
    MyPuzzle = Puzzle(8, int(8 * 8 * 0.6))
```
[4]

(b) Many languages, such as C# and Java, allow methods of the same name but with different signatures to be defined in the same class.

State the name of this OOP technique. [1]

2 This question refers to the entire code. Throughout the code, several numbers have been hard-coded (see two examples below).

```
Example 1:
    MyPuzzle = Puzzle(8, int(8 * 8 * 0.6))
Example 2:
    if random.randrange(1, 101) < 90:
```

(a) State two reasons why constants would be more appropriate. [2]

(b) The code is written using the object-oriented paradigm.

Discuss two advantages of this over the traditional structured approach to programming. [4]

3 This question refers to the CheckforMatchWithPattern method of the Puzzle class. The current pattern string used to match the Q pattern in the puzzle is: `"QQ**Q**QQ"`

(a) Explain how a successful match is determined. [2]

(b) Explain how an unsuccessful match is determined. [1]

(c) Explain how players are prevented from placing any more letter Qs in the same 3 × 3 grid once a successful match has been determined. [1]

4 This question refers to the MatchesPattern method of the Pattern class.

The pattern matching currently uses an iterative statement; however, regular expressions (regex) could be used instead. The regex for the T pattern would be TTT..T..T where the . symbol is used to match any character.

State the regex for the Q pattern. [1]

5 This question refers to the Cell and BlockedCell classes.

(a) BlockedCell inherits from Cell. Define the term 'inherits'. [2]

(b) Describe two advantages of BlockedCell inheriting from Cell rather than just having a Cell with the symbol '@'. [4]

(c) Define the term 'override' in the context of object-oriented programming. [2]

6 This question refers to the AttemptPuzzle and GetCell methods of the Puzzle class. In each of these methods is an example of exception handling.

(a)
```
Valid = False
  while not Valid:
    try:
      Row = int(input("Enter row number: "))
      Valid = True
    except:
      pass
```

Describe, in detail, the purpose of the code snippet above from the AttemptPuzzle method. Include in your answer an explanation of how the try…except construct is used in this snippet. [4]

(b) The GetCell method sometimes raises an exception.

Describe the effect of raising an exception. [2]

7 This question refers to the DisplayPuzzle method of the Puzzle class.

(a) Identify one use of polymorphism of the Cell and BlockedCell classes in this method. [1]

(b) Define the term 'polymorphism'. [3]

8 This question refers to the GetCell method of the Puzzle class.

(a) Programming to the interface and not the implementation is one advantage of encapsulation.

Describe how the code for GetCell demonstrates encapsulation. [3]

(b) Describe the effect of the method being called with the arguments Row=2 and Column=10 using the standard Grid size of 8. [2]

9 This question refers to the class Cell. One protected attribute of Cell is Symbol, which contains the current contents of the Cell.

(a) Empty cells contain the empty string "" yet are displayed on screen as "-".

Describe the route of execution of code that achieves this, starting with a call to GetSymbol. [3]

(b) The class Cell also contains a private attribute.

Explain the difference between a protected attribute and a private attribute. [2]

10 This question refers to the class Puzzle.

(a) The private attribute Grid is a list of Cell references.

Describe one difference between a list and an array. [1]

(b) One alternative for storing the Grid would have been to use a two-dimensional array of Cell references.

Compare and contrast using a two-dimensional array with using a (one-dimensional) list of Cell references to store the Grid. [4]

11 This question refers to the CheckforMatchWithPattern method of the Puzzle class. Assuming that the current puzzle looks as follows:

	1	2	3	4	5
5	X		X		X
4		X		X	
3	X				X
2		X		X	
1	X		X		X

(a) A score of 10 will be awarded for placing an X symbol in the grid square [3,3].

Describe the code logic that awards only 10 for this, but limit your description to the CheckforMatchWithPattern method. [2]

(b) Earlier in the game, the puzzle looked as follows:

	1	2	3	4	5
5	X		X		X
4		X		X	
3	X				
2		X		X	
1			X		

State a list of four moves in order, each of which places an additional X symbol on the Grid that will result in a total score of 40. [2]

12 A new feature is being added into the Puzzle that will calculate the maximum possible score that can be achieved from any current Grid. This algorithm will require each pattern to be placed in each possible section of the board in which it could fit in combination with each other pattern. For example, on the 5 × 5 board shown below:

@				
		@	@	

each 3 × 3 grid would be checked in turn, starting in the top left to see where the Q pattern could be placed, and the answers would be:

		Q	Q	
@		Q	Q	
Q	Q	@	@	Q
Q	Q			
		Q		

similarly for T as shown below:

T	T	T		
@	T			
	T	@	@	

and then finally for X, which shows that there are no possible pattern positions.

The algorithm would then place a Q pattern on the grid and try to place all possible combinations of other patterns (only the second Q pattern is possible). It would then place a T pattern on the Grid and try to place all other possible combinations of patterns (none possible). The greatest number of patterns that could be placed is two, resulting in a score of 20.

(a) State the time complexity of this algorithm where n is the maximum number of patterns that can be placed. [1]

(b) Explain how you came to your answer for part a). [2]

(c) Define what a tractable problem is. [2]

13 This question refers to the DisplayPuzzle method of the Puzzle class. The first selection statement which contains an iterative statement displays the column numbers and a horizontal line on the Console.

Describe in detail how the second iterative structure formats the rest of the Grid that is displayed on the Console.

Ensure that your description explains how the use of integer division affects the output. [6]

14 This question refers to the constructor of the BlockedCell class.

(a) Explain the purpose of the keyword super(). [1]

(b) Describe the meaning of a public attribute. [1]

(c) Explain why the use of private and protected attributes provides better encapsulation than public attributes. [3]

15 This question involves the creation of a new algorithm. A new bonus score is being introduced for any player who manages to achieve the following 4 × 4 pattern in their grid.

Q	Q	Q	Q
T	T	X	X
T	X	X	T
T	T	X	X

The bonus will be 50 points. Write an algorithm that can be used to check for this pattern. The algorithm can be in pseudocode, flowchart or structured English and should return 0 if the pattern is not found and 50 if the pattern is found in the grid. The algorithm should be for a function called CheckForSpecialPattern. You may wish to refer to the CheckforMatchWithPattern method for suitable logic. [6]

END OF QUESTIONS