## Programming Questions (Solutions) **MORSE CODE**

| **Q** | **Answer/Guidance** | **Marks** |
| --- | --- | --- |
| 1a | EMPTYSTRING | **1** |
| 1b | GetNextSymbol // GetNextLetter | **1** |
| 1c | GetNextSymbol // GetNextLetter  | **1** |
| 1d | LetterEnd // ProgramEnd | **1** |
| 1e | Dash // Letter // Dot // MorseCode | **1** |
| 1f | Dash // Dot | **1** |
| 1g | len // input | **1** |
| 1h | GetNextSymbol | **1** |
| 2 | 1 mark for each of the following:* (String) variable (FileName) initialised to user input
* (Variable) FileHandle assigned to specified file opened for read only access using the open function
* Transmission variable set to first line of the file
* File is closed
 | **4** |
| 3 | 1 mark for each of the following:* (Repeatedly) prompt the user / accept user input
* until X is entered / loop terminates at X
 | **2** |
| 4 | 1 mark for each of the following **(max 3)**:* Integer array
* Contains pointers
* Indicates which element to move to next...
* ... if the next Morse signal is a dash
 | **3** |
| 5 | 1 mark for each of the following:* Initially set to the first character in Transmission
* As spaces are removed, it points to the new first character
 | **2** |
| 6 | 1 mark for each of the following:* (Integer) variable LastChar set to the index of the last character in Transmission
* Using the built-in function len() to get the length of the Transmission
* Loop repeats while LastChar / last character is a space
* If the last character is a space, remove it (from Transmission)
* Decrement LastChar / integer variable
* Return Transmission, with all spaces removed from end/right
 | **6** |
| 7 | 1 mark for each of the following **(max 3)**:* Gets ASCII value of PlainTextCharacter
* Gets ASCII value of A / Gets value 65
* Subtracts ASCII value of A / 65 from ASCII value of PlainTextCharacter and adds 1
* If PlainTextLetter is A, Index is 1 (for example)
 | **3** |
| 8 | 1 mark for each of the following:* except: block executed if try: block fails to execute correctly / generates exception
* File name mistyped // file not found // error reading file // error/exception in StripLeadingSpaces // error/exception in StripTrailingSpaces // Transmission/EOL not being a valid string
 | **2** |
| 9 | 1 mark for each of the following **(max 2)**:* Constants won’t be accidentally changed
* By being at the start of the code, the code is easier to read/understand (*must be clear that this is for the benefit of the human, not the computer*)
* No need to remember (precise) values // constant names more memorable than values // code is more readable
 | **2** |
| 10 | 1 mark for each of the following **(max 3)**:* It is used to remove the first item in a list
* It Is used here to trim the first character/space from the Transmission
* By treating the string as a list
* Is called repeatedly if multiple spaces exist
 | **3** |
| 11 | 1 mark for StripLeadingSpaces instance:* If the length of Transmission s zero

1 mark for GetTransmission instance:* If there is a file error (accept any error relating to code in the try: block) // if the except: block executes // if the try: block fails / generates an error/exception

3 marks for GetNextSymbol instance:* If the symbol is not a dot...
* ... not a dash / minus sign...
* ... not a space
 | **5** |
| 12 | 1 mark for each of the following:* CodedLetterLength variable set to the length of the sequence to be decoded (4)
* for loop to run four times
* Symbol initially set to the first symbol in the sequence to be decoded (- dash)
* Pointer set to 20
* Symbol then set to dot (on next iteration)
* Pointer set to 14
* Pointer set to 4 (on next iteration)
* Symbol then set to dash (on next iteration)
* Pointer set to 24
* X retrieved from Letter array / X returned (*only credit this mark if there is an attempt to parse the arrays*)
 | **10** |
| **TOTAL MARKS** | **50** |