**Godalming College**

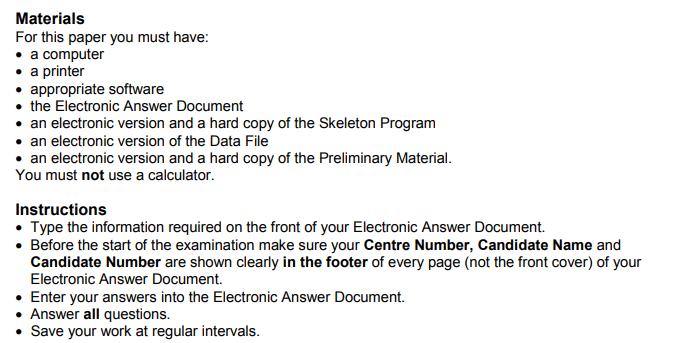
**Computer Science Department**



Easter Benchmark

April 2021

1 Hour Maximum score is 55 marks.



Do not write on this paper,

Do not share answers with anyone else  
Complete this paper in the time allowed.

If you would find it easier and are able, please print this Exam Paper out

**Section B**

You are advised to spend no more than **20 minutes** on this section.

Enter your answers to **Section B** in your Electronic Answer Document. You **must save** this document at regular intervals.

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

Refer **either** to the **Preliminary Material** issued with this question paper **or** your electronic copy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **0** | **4** |  |  | State the name of an identifier for: |
|  |  | **.** |  | a local variable that is used to store a Boolean value. |
| **0** | **4** | **1** |

**[1 mark]**

**0** **4** **.** **2** a user-defined subroutine that has one parameter.

**[1 mark]**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **0** | **4** | **.** | **3** | a user-defined subroutine that returns a String value. |

**[1 mark]**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **0** | **5** |  |  |  |
|  |  | **.** |  | A constant called ‘EMPTYSTRING’ is used to hold “”. |
| **0** | **5** | **1** |

State **one** advantage of using named constants for constant values.

**[1 mark]**

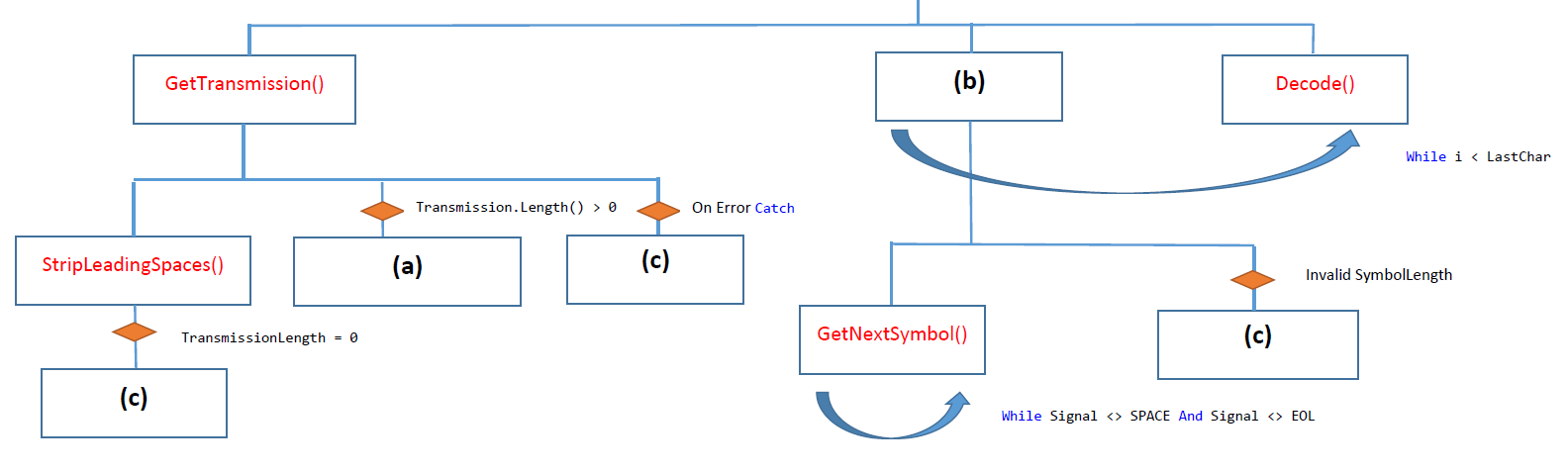
The GetNextLetter subroutine is called as part of the ReceiveMorseCode subroutine to obtain the string of dots and dashes corresponding to each morse code character. GetNextLetter uses a WHILE loop to check for the end of a coded letter.

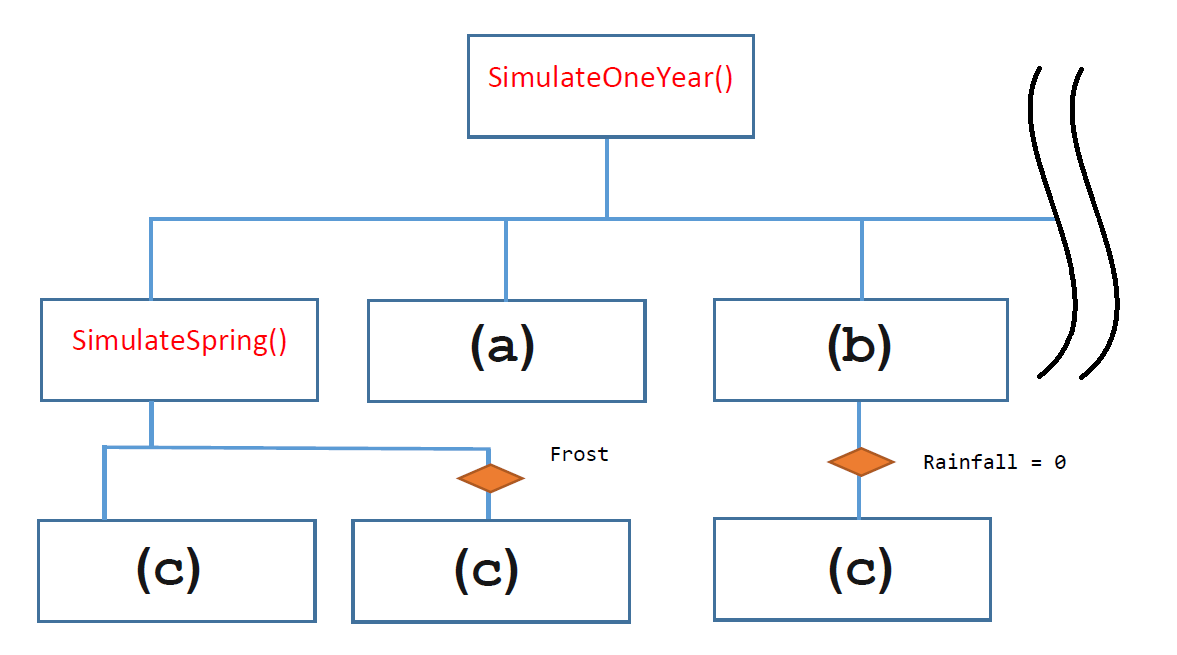
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **.** |  | Explain why this is an appropriate loop structure. |
| **0** | **5** | **2** |
|  |  | **.** |  | **[2 marks]** |
|  |  |  | There are 3 conditions which terminate the above loop. State these conditions and explain in words the significance of each. |
| **0** | **5** | **3** |
|  |  |  |  | **[3 marks]** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **.** |  | The GetTransmission subroutine uses exception handling. Explain what is meant by |
| **0** | **5** | **4** |
|  |  |  |  | exception handling, how it is used and what would happen if the user enters a file name that does not exist. |
|  |  |  |  |
|  |  |  |  | **[3 marks]** |

|  |  |  |
| --- | --- | --- |
| **0** | **6** | **Figure 5** shows an incomplete hierarchy chart for part of the **Skeleton Program**. |

**Figure 5**





With reference to the **Skeleton Program** and **Figure 5**, answer the questions below.

**0** **6** **.** **1** What should be written in box **(a)** in **Figure 5**? **[1 mark]**

**0** **6** **.** **2** What should be written in box **(b)** in **Figure 5**? **[1 mark]**

**0** **6** **.** **3** What should be written in boxes **(c)** in **Figure 5**? **[1 mark]**

A structured programming approach has been used in the production of the Skeleton Program.

**0** **6** **.** **4** Explain what is meant by a structured programming approach.

**[2 marks]**

|  |  |  |
| --- | --- | --- |
| **0** | **7** |  |

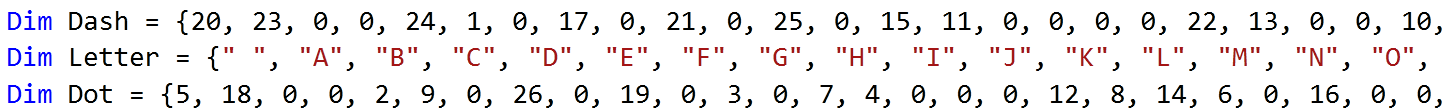
There is a variable called MorseCodeString in the subroutine SendMorseCode.

There is also a different variable called MorseCodeString in the subroutine ReceiveMorseCode.

**0** **7** **.** **1** Explain why these two different variables can have the same identifier.

**[2 marks]**

**0 7. 2** The followingcode extract shows how part of the decode tree for the alphabetical characters is stored in the program:



Note that all arrays are indexed from zero.

Complete the following trace table to ‘decode’ the Morse character “. . - . “

|  |  |  |  |
| --- | --- | --- | --- |
| Morse Character | Dot | Dash | Alphabetical  Character |
| . | 5 |  |  |
| . |  |  |  |
| - |  |  |  |
| . |  |  |  |

**[4 marks]**

**Section C**

You are advised to spend no more than **40 minutes** on this section.

Enter your answers to **Section C** in your Electronic Answer Document. You **must save** this document at regular intervals.

These questions require you to load the **Skeleton Program** and to make programming changes to it.

|  |  |  |
| --- | --- | --- |
| **0** | **8** | This question refers to the subroutine SendReciveMessages |
|  | | **What you need to do:** |
| **Task 1** |
| Modify the subroutine SendReciveMessages so that it displays the total number of dots and dashes in the Morse code alphabet before the Main Menu. The first time the program is run ( not need for it to be displayed every time the main menu is displayed |
| Figure 1 |

The subroutine must calculate the number of dots and dashes. Only 1 mark will be credited for a “hard coded” solution .

**Task 2**

Test that the changes you have made work by conducting the following test:

* Run the Program
* View the message as per figure 1
* Exit the program

**1**

**2**

**Evidence that you need to provide**

Include the following in your Electronic Answer Document.

**0 8 .**

Your amended PROGRAM SOURCE CODE for the subroutine

SendReceiveMessages.

**[5 marks]**

**0 8 .**

SCREEN CAPTURE(S) showing the requested test.

**[1 mark]**

|  |  |  |
| --- | --- | --- |
| **0** | **9** | This question will add the functionality of the program  . |
|  | | Joe wants to learn Morse code and would like a mini quiz to test his progress.  He would like to be presented with a Morse Code Letter and prompted to guess which letter of the alphabet it represents.  **What you need to do:**  Add a menu option that calls new subroutine that will display a morse letter and ask the user to guess which letter of the alphabet it is. |
| **Task 1** |
| Modify the DisplayMenu and SendReceiveMessages subroutines So that there is an option “Q” that will call a new subroutine called “MorseWang” |
| **Task 2**  Create a new Subroutine Called MorseWang.  MorseWang will need to accept 2 parameters: The array of MorseCodes and the array of Letters  It should present the user with a Random Morse letter and display the Following Prompt: ***Please guess which upper case letter this Morse code represents:***  The program should then assess the Users input and Respond with one of the following options:   * Congratulations that’s MORSEWANG * :( you got it wrong.   After the question the program should return to the main menu |
| Test that your amended program code works by conducting the following test: |

* + start the Program
  + enter Q for the mini quiz
  + enter the correct answer (use the pre-lim material to help you)
  + Bask in the glory of your right answer
  + enter Q for the mini quiz
  + enter an upper case letter that is the wrong answer)
  + Try not to feel too dissapointed

**1**

**2**

**Evidence that you need to provide**

Include the following in your Electronic Answer Document.

**0 9 .**

Your amended PROGRAM SOURCE CODE for the subroutines, **SendReceiveMessages**. And **MorseWang [5 marks]**

**0 9 .**

SCREEN CAPTURE(S) showing the above tests..

**[1 mark]**

**Turn over for the next question**

**1**

**0**

This question refers to **SendMorseCode**.

Currently, the program converts user-input plaintext into Morse code and displays the Morse code on the screen.

Modify the code so that each new Morse code message generated in SendMorseCode is converted to a transmission format (e.g. =⌂===⌂⌂⌂=⌂=⌂=) and displayed in this format on the console. (NB ⌂ represents a space character)

This Transmission format message needs to be saved to a File named “output.txt” A message should be displayed in the screen stating: “***Message Sent Successfully”***

**What you need to do: Task 1**

Modify the **SendMorseCode** subroutine so that it outputs the message to the screen in transmission format

**Task 2**

Further modify the **SendMorseCode** subroutine so that it writes the message in transmission format to a File called “output.txt”

**Test 1**

Test that the subroutine **SendMorseCode** works by conducting the following test:

* Run the Morse Code Program
* Enter “S” to select the “Send message” Option
* Enter the following message:  
  **I EAT TEA** and press enter
* Enter “R” to select “Receive Message”
* Type “Output.txt” to open the newly created file

**1**

**2**

**3**

**Evidence that you need to provide**

Include the following in your Electronic Answer Document.

**1 0 .**

Your PROGRAM SOURCE CODE for the subroutine **SendMorseCode** .

**[8 marks]**

**1 0 .**

For the first run of the simulation: SCREEN CAPTURE(S) showing console displaying **test 1**

**[1 mark]**

**1 0 .**

For the second run of the simulation: SCREEN CAPTURE(S) showing the date file output.txt opened in the notepad application

**[1 mark]**

**END OF QUESTIONS**