

|  |  |
| --- | --- |
| Name |  |
| ZigZag Education supporting  AS AQA Computer Science Paper 1  Summer 2018  MORSE CODE  Electronic Answer Document (EAD)  **Instructions**   Enter your name in the box at the top of this page   Answer **all** questions by entering your answers into this document   Remember to **save** this document regularly   Save and print this document and any additional pages   Answer **all** questions   The marks available for each question are shown in brackets   You will need:  □ access to a computer  □ access to a printer  □ access to appropriate software  □ electronic copies of the required skeleton code  □ EAD (Electronic Answer Document)  Total marks: | |
|

|  |
| --- |
| Written Questions  Answer all questions.  Remember to save this document regularly. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Q** | **Answer** | | *Mark  (leave blank)* |
| 1 | (a) | SPACE |  |
| (b) | GetNextSymbol |  |
| (c) | GetNextLetter |  |
| (d) | LetterEnd |  |
| (e) | Letter |  |
| (f) | Dot |  |
| (g) | len |  |
| (h) | I, Symbol |  |
| 2 | To Get Filename Then Open And Read The File To Get The Morse Code. | |  |
| 3 | To Ensure The Correct Input Is Entered And To Allow The User To Choose An Option. | |  |
| 4 | It Is A List To Store Index Values Of Dots To Work In Parallel With The Dash List. | |  |
| 5 | It Gets The First Letter Of The String In Order To Check If It Is A Space | |  |
| 6 | Assigns Last Character Index Of The String.  Checks If The Last Character Is A Space Until There Is No Spaces  Decreases LastChar Variable By 1  Removes The Last Character Of The String Transmission  Then Returns The String Transmission After It Is Completed | |  |
| 7 | Sets the index based on the ord value of the letter then subtracts ord(“A”) to get an index inside of the letter list. | |  |
| 8 | To manage error handling and set to empty string. | |  |
| 9 | Variables are global, You can easily change the values. | |  |
| 10 | Starts from the second place of the input to the end. | |  |
| 11 | No signal received, No transmission found, Non-standard symbol received. | |  |
| 12 | It will cross-reference the lists using index values to return the correct letter. | |  |

|  |
| --- |
| Programming Tasks  Answer all questions.  Remember to save this document regularly. |

|  |  |  |
| --- | --- | --- |
| **Q** | **Answer** | *Mark  (leave blank)* |
| 1 | def GetMenuOption():  MenuOption = EMPTYSTRING  while len(MenuOption) != 1:  MenuOption = input("Enter your choice: ").upper()  if MenuOption != "R" and MenuOption != "S" and MenuOption != "X":  print("Invalid choice, please choose a letter from the menu:")  DisplayMenu()  return MenuOption |  |
| 2 | FULLSTOP = "."  Letter = [' ','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z', "."]  MorseCode = [' ','.-','-...','-.-.','-..','.','..-.','--.','....','..','.---','-.-','.-..','--','-.','---','.--.','--.-','.-.','...','-','..-','...-','.--','-..-','-.--','--..', ".-.-.-"] |  |
| 3 | def printMorseCodeSymbols(Letter, MorseCode):  for i in range(len(Letter)):  print(f"{Letter[i]} | {MorseCode[i]}")  def DisplayMenu():  print()  print("Main Menu")  print("=========")  print("R - Receive Morse code")  print("S - Send Morse code")  print("P - Print Morse code symbols")  elif MenuOption == "P":  printMorseCodeSymbols(Letter, MorseCode) |  |
| 4 | def transmitMorseCode(MorseCode):  result = SendMorseCode(MorseCode)  Data = ""  for i in result:  if i == ".":  Data += "= "  elif i == "-":  Data += "=== "  if i == " ":  Data += " "  fileName = input("Enter name of file: ")  with open(fileName, "w+") as file:  file.write(Data)  file.close() |  |
| 5 | if *CodedLetter* not in *MorseCode*:      print(f"Invalid Symbol ({*CodedLetter*}) recieved")      return "\*" |  |
| 6 | if ".txt" not in FileName:      FileName += ".txt" |  |
| 7 | def ConvertMorseCode(*MorseCode*, *Letter*):    Message = input("Enter a message in Morse Code: ")    Message = Message.split(" ")    Output = ""    for i in Message:      if i in *MorseCode*:        Index = list.index(*MorseCode*, i)        Output += *Letter*[Index]      else:        print("Invalid Symbol")        Output += "\*"    print(Output) |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
| 11 |  |  |
| 12 |  |  |