

Teaching guide: Programming challenge 3

Guess the word

Programming is a fundamental skill required for success in GCSE Computer Science. This programming challenge is designed to develop students' programming skills.

In the paper 1 exam, students will be required to design, write, test and refine program code in either C#, Python (version 3) or VB.Net.

To develop their programming skills, students should have sufficient practical experience of:

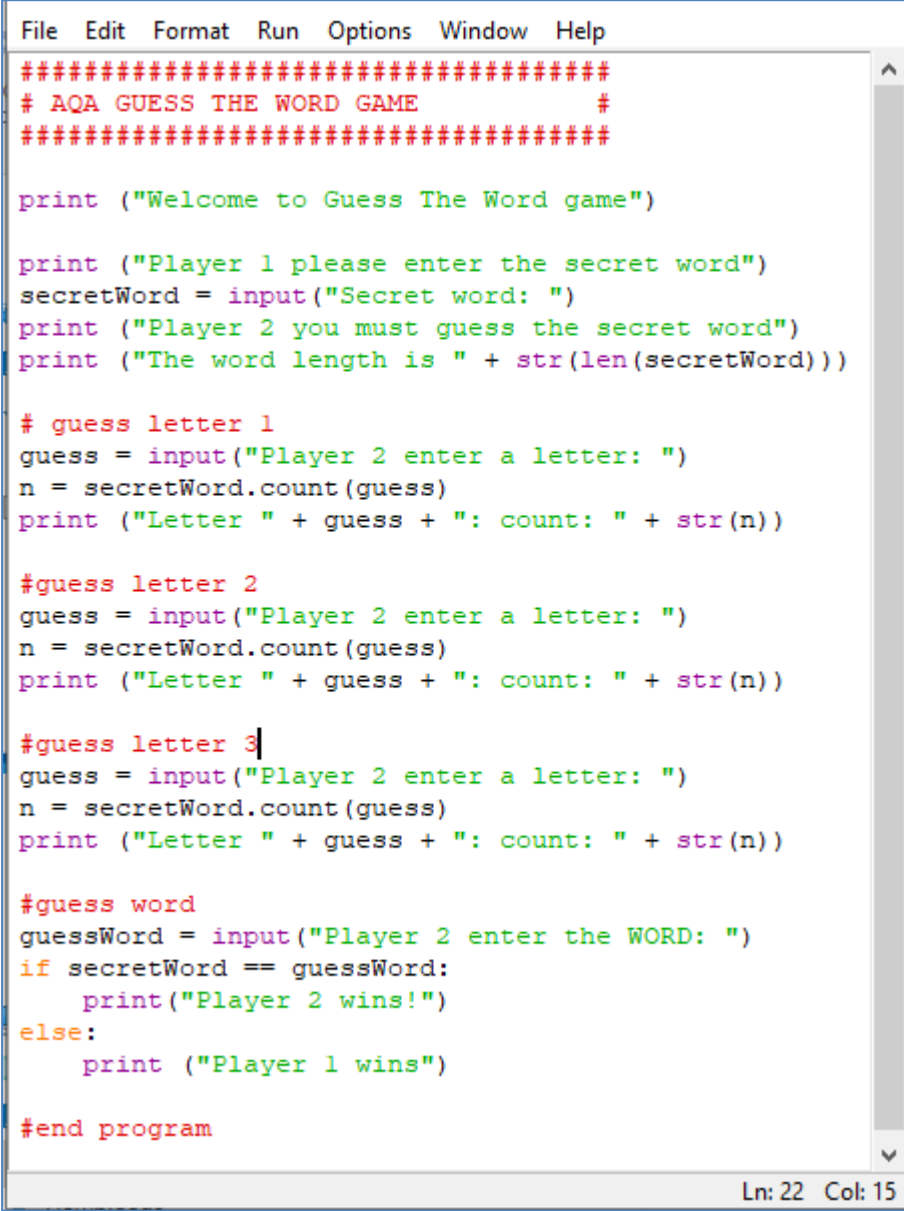
- structuring programs into modular parts with clear documented interfaces to enable them to design appropriate modular structures for solutions
- including authentication and data validation systems/routines within their computer programs
- writing, debugging and testing programs to enable them to develop the skills to articulate how programs work and argue using logical reasoning for the correctness of programs in solving specified problems
- designing and applying test data (normal, boundary and erroneous) to the testing of programs so that they are familiar with these test data types and the purpose of testing
- refining programs in response to testing outcomes.

Programming Challenge 3 – Guess the word

Exercise 1

This exercise can be solved using either an iteration structure or by repeating the same code multiple times. It is an opportunity to explain program efficiency to students. The basic example solution in **Figure 1** shows a non-looping solution. **Figure 2** shows a version using iteration.

Figure 1



```
File Edit Format Run Options Window Help
#####
# AQA GUESS THE WORD GAME #
#####

print ("Welcome to Guess The Word game")

print ("Player 1 please enter the secret word")
secretWord = input("Secret word: ")
print ("Player 2 you must guess the secret word")
print ("The word length is " + str(len(secretWord)))

# guess letter 1
guess = input("Player 2 enter a letter: ")
n = secretWord.count(guess)
print ("Letter " + guess + ": count: " + str(n))

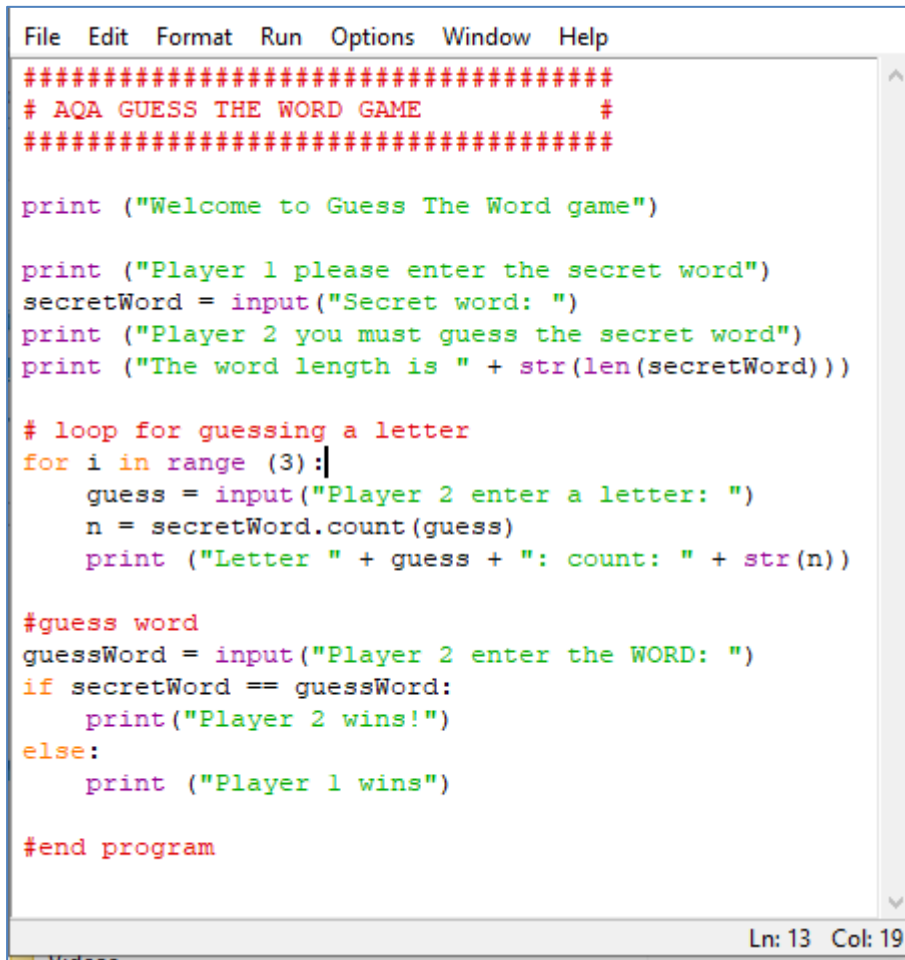
#guess letter 2
guess = input("Player 2 enter a letter: ")
n = secretWord.count(guess)
print ("Letter " + guess + ": count: " + str(n))

#guess letter 3
guess = input("Player 2 enter a letter: ")
n = secretWord.count(guess)
print ("Letter " + guess + ": count: " + str(n))

#guess word
guessWord = input("Player 2 enter the WORD: ")
if secretWord == guessWord:
    print("Player 2 wins!")
else:
    print ("Player 1 wins")

#end program
Ln: 22 Col: 15
```

Figure 2

A screenshot of a Python IDE window. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The main area contains Python code for a word-guessing game. The code starts with a header in red: '#####', '# AQA GUESS THE WORD GAME', and '#####'. It then prints a welcome message, asks for a secret word, and asks for a guess. A loop for guessing a letter is shown, followed by a check for a win or loss. The status bar at the bottom right shows 'Ln: 13 Col: 19'.

```
File Edit Format Run Options Window Help
#####
# AQA GUESS THE WORD GAME #
#####

print ("Welcome to Guess The Word game")

print ("Player 1 please enter the secret word")
secretWord = input("Secret word: ")
print ("Player 2 you must guess the secret word")
print ("The word length is " + str(len(secretWord)))

# loop for guessing a letter
for i in range (3):|
    guess = input("Player 2 enter a letter: ")
    n = secretWord.count(guess)
    print ("Letter " + guess + ": count: " + str(n))

#guess word
guessWord = input("Player 2 enter the WORD: ")
if secretWord == guessWord:
    print("Player 2 wins!")
else:
    print ("Player 1 wins")

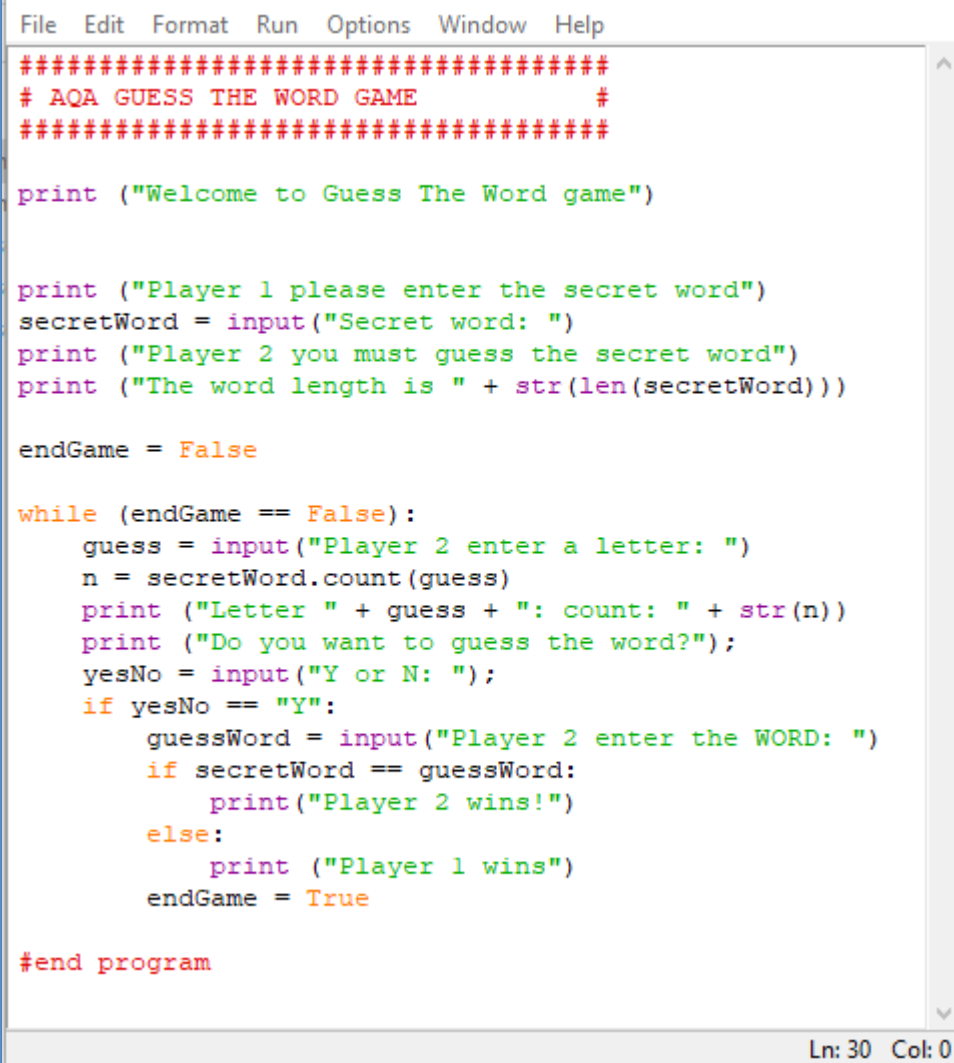
#end program

Ln: 13 Col: 19
```

Exercise 2

This exercise requires replacing a definite loop with an indefinite loop and adds a nested IF statement.

Figure 3



```
File Edit Format Run Options Window Help
#####
# AQA GUESS THE WORD GAME #
#####

print ("Welcome to Guess The Word game")

print ("Player 1 please enter the secret word")
secretWord = input("Secret word: ")
print ("Player 2 you must guess the secret word")
print ("The word length is " + str(len(secretWord)))

endGame = False

while (endGame == False):
    guess = input("Player 2 enter a letter: ")
    n = secretWord.count(guess)
    print ("Letter " + guess + ": count: " + str(n))
    print ("Do you want to guess the word?");
    yesNo = input("Y or N: ");
    if yesNo == "Y":
        guessWord = input("Player 2 enter the WORD: ")
        if secretWord == guessWord:
            print("Player 2 wins!")
        else:
            print ("Player 1 wins")
        endGame = True

#end program

Ln: 30 Col: 0
```

Exercise 3

Validation can be added in functions. There is an opportunity for students to practice functions with more than one parameter.

Figure 4

```
#####
# AQA GUESS THE WORD GAME      #
#####
def validateLetter(guess, guessedLetters):

    while True:
        if guess.isalpha() and len(guess) == 1 and guess not in guessedLetters:
            return guess
        else:
            print ("Your letter must be a single letter a to z, and not guessed before")
            guess = input("Player 2 enter a letter: ")

def validateWord(secretWord):
    while True:
        if secretWord.isalpha() and len(secretWord) > 5 and len(secretWord) < 13:
            return secretWord
        else:
            print ("Your word must only contain letters and have length between 6 and 12")
            secretWord = input("Secret word: ")

#####
## Main program #
#####
print ("Welcome to Guess The Word game")

print ("Player 1 please enter the secret word")
secretWord = validateWord(input("Secret word: "))

print ("Player 2 you must guess the secret word")
print ("The word length is " + str(len(secretWord)))

endGame = False
guessedLetters = ""
nTotal = 0
while (endGame == False):
    guess = validateLetter(input("Player 2 enter a letter: "), guessedLetters)
    guessedLetters += guess
    print(guessedLetters)
    n = secretWord.count(guess)
    print ("Letter " + guess + ": count: " + str(n))
    nTotal += n

    ## win validation
    if nTotal == len(secretWord):
        print("you have guessed all the letters, now guess the word")
        yesNo = "Y"
    else:
        print ("Do you want to guess the word?");
        yesNo = input("Y or N: ");

    if yesNo.lower() == "y":
        guessWord = input("Player 2 enter the WORD: ")
        if secretWord == guessWord:
            print("Player 2 wins!")
        else:
            print ("Player 1 wins")
            endGame = True

#####
#end program
```

Ln: 19 Col: 47

Exercise 4

This exercise can be done on any version of the task. The example below shows code added to **Figure 3**. This gives students the opportunity to use a list/array and to iterate through the array to replace one value with another, to give a partly guessed word that can be output, to help Player 2. Note: this version outputs the contents of the array, not the string with replaced letters. Students should be encouraged to think how they can convert an array to a string. Some students may realise that this functionality negates the win validation in Exercise 3.

Figure 5

```
File Edit Format Run Options Window Help
#####
# AQA GUESS THE WORD GAME #
#####

print ("Welcome to Guess The Word game")

print ("Player 1 please enter the secret word")
secretWord = input("Secret word: ")
print ("Player 2 you must guess the secret word")
print ("The word length is " + str(len(secretWord)))
partWord = []
for i in range(0, len(secretWord)):
    partWord.append("_")

endGame = False

while (endGame == False):
    guess = input("Player 2 enter a letter: ")
    n = secretWord.count(guess)
    print ("Letter " + guess + ": count: " + str(n))

    for i in range(0, len(secretWord)):
        if guess == secretWord[i]:
            partWord[i] = guess

    print("The partly guessed word is:" + str(partWord))

    print ("Do you want to guess the word?");
    yesNo = input("Y or N: ");
    if yesNo == "Y":
        guessWord = input("Player 2 enter the WORD: ")
        if secretWord == guessWord:
            print("Player 2 wins!")
        else:
            print ("Player 1 wins")
        endGame = True

#end program

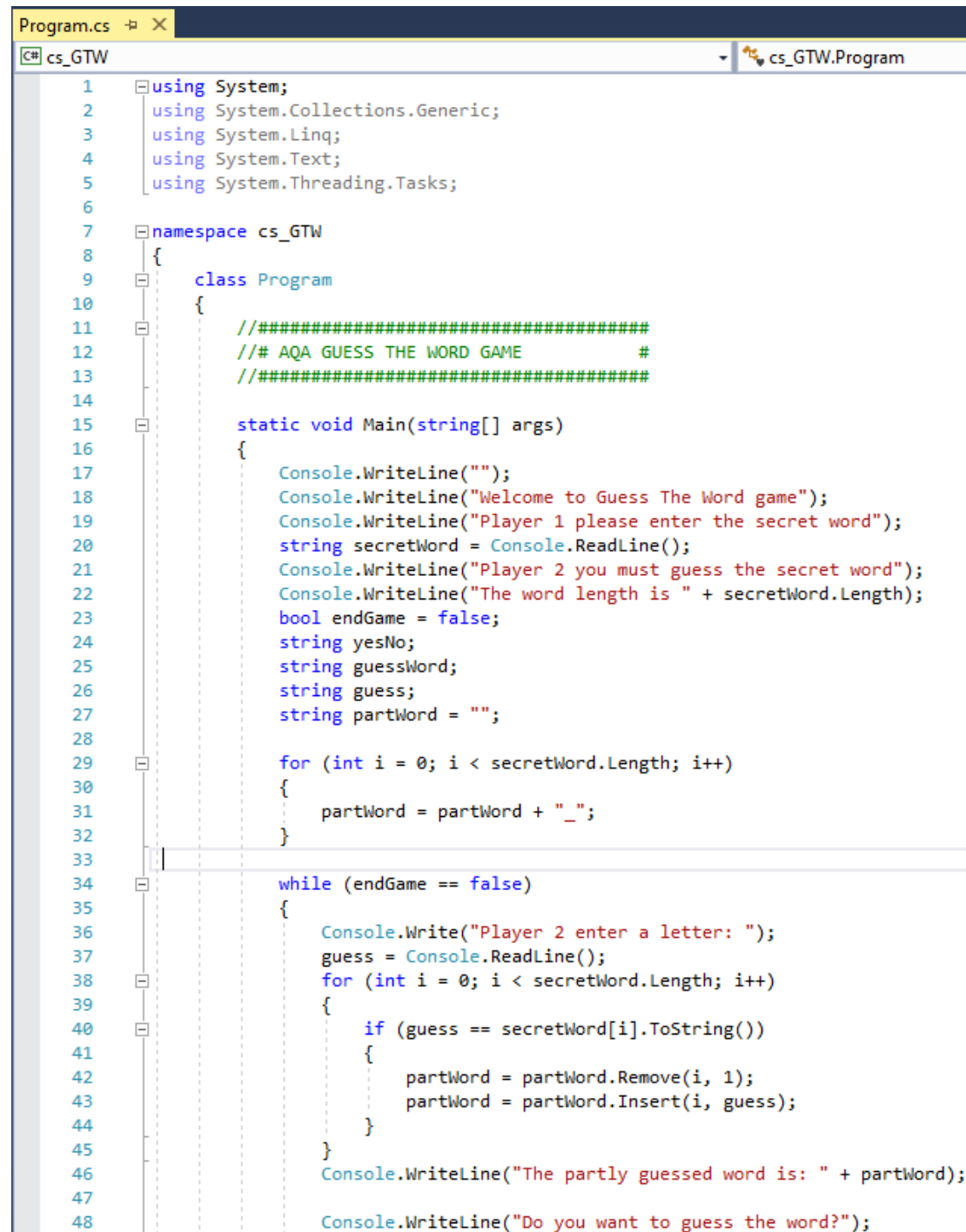
Ln: 15 Col: 0
```

Extensions 1 and 2

Extension tasks will stretch the more able students, using techniques already used, but with the emphasis on designing a complete solution. Students could be encouraged to explore the Try Catch functionality in Python or the equivalent structures in their chosen language.

Appendix C# and VB.NET versions for Exercise 4

C#



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace cs_GTW
8 {
9     class Program
10    {
11        //#####
12        /// AQA GUESS THE WORD GAME #
13        //#####
14
15        static void Main(string[] args)
16        {
17            Console.WriteLine("");
18            Console.WriteLine("Welcome to Guess The Word game");
19            Console.WriteLine("Player 1 please enter the secret word");
20            string secretWord = Console.ReadLine();
21            Console.WriteLine("Player 2 you must guess the secret word");
22            Console.WriteLine("The word length is " + secretWord.Length);
23            bool endGame = false;
24            string yesNo;
25            string guessWord;
26            string guess;
27            string partWord = "";
28
29            for (int i = 0; i < secretWord.Length; i++)
30            {
31                partWord = partWord + "_";
32            }
33
34            while (endGame == false)
35            {
36                Console.Write("Player 2 enter a letter: ");
37                guess = Console.ReadLine();
38                for (int i = 0; i < secretWord.Length; i++)
39                {
40                    if (guess == secretWord[i].ToString())
41                    {
42                        partWord = partWord.Remove(i, 1);
43                        partWord = partWord.Insert(i, guess);
44                    }
45                }
46                Console.WriteLine("The partly guessed word is: " + partWord);
47
48                Console.WriteLine("Do you want to guess the word?");
```

```

49     Console.Write("Y or N: ");
50     yesNo = Console.ReadLine();
51     if (yesNo == "Y")
52     {
53         Console.Write("Player 2 enter the WORD: ");
54         guessWord = Console.ReadLine();
55
56         if (secretWord == guessWord)
57         {
58             Console.WriteLine("Player 2 wins!");
59         }
60         else
61         {
62             Console.WriteLine("Player 1 wins!");
63             endGame = true;
64         }
65     }
66 }
67 }
68 }
69 }
70 }

```

VB.NET

```

Module1.vb
vb_GTW
Module1
1  Module Module1
2
3  Sub Main()
4      '#####
5      '# AQA GUESS THE WORD GAME      #
6      '#####
7
8      Console.WriteLine("")
9      Console.WriteLine("Welcome to Guess The Word game")
10     Console.WriteLine("Player 1 please enter the secret word")
11     Dim secretWord As String = Console.ReadLine()
12     Console.WriteLine("Player 2 you must guess the secret word")
13     Console.WriteLine("The word length is " + Str(Len(secretWord)))
14     Dim endGame As Boolean = False
15     Dim yesNo As String
16     Dim guessWord As String
17     Dim guess As String
18     Dim partWord As String = ""
19
20     For i As Integer = 0 To Len(secretWord)
21         partWord = partWord + "_"
22     Next
23
24     While endGame = False
25         Console.Write("Player 2 enter a letter: ")
26         guess = Console.ReadLine()
27

```



```
28 For i As Integer = 0 To Len(secretWord) - 1
29     If guess = secretWord(i) Then
30         Console.WriteLine("guess correct ")
31         partWord = partWord.Remove(i, 1)
32         partWord = partWord.Insert(i, guess)
33     End If
34 Next
35
36 Console.WriteLine("The partly guessed word is: " + partWord)
37
38 Console.WriteLine("Do you want to guess the word?")
39 Console.Write("Y or N: ")
40 yesNo = Console.ReadLine()
41
42 If yesNo = "Y" Then
43     Console.Write("Player 2 enter the WORD: ")
44     guessWord = Console.ReadLine()
45     If secretWord = guessWord Then
46         Console.WriteLine("Player 2 wins!")
47     Else
48         Console.WriteLine("Player 1 wins!")
49         endGame = True
50     End If
51 End If
52 End While
53 End Sub
54
55 End Module
56
```