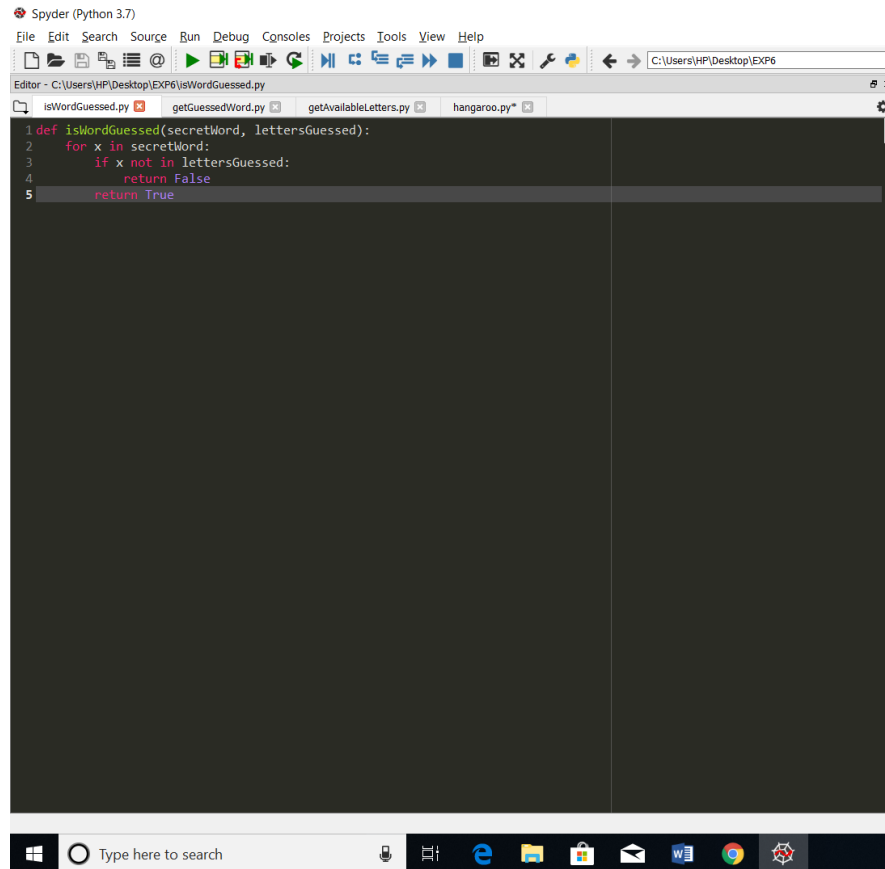
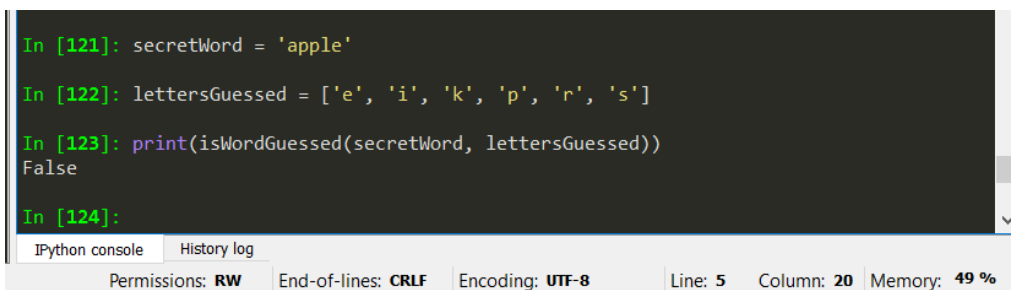


Malabanan, Angelo Quim C.
2ECE-A

1.)



```
1 def isWordGuessed(secretWord, lettersGuessed):
2     for x in secretWord:
3         if x not in lettersGuessed:
4             return False
5     return True
```

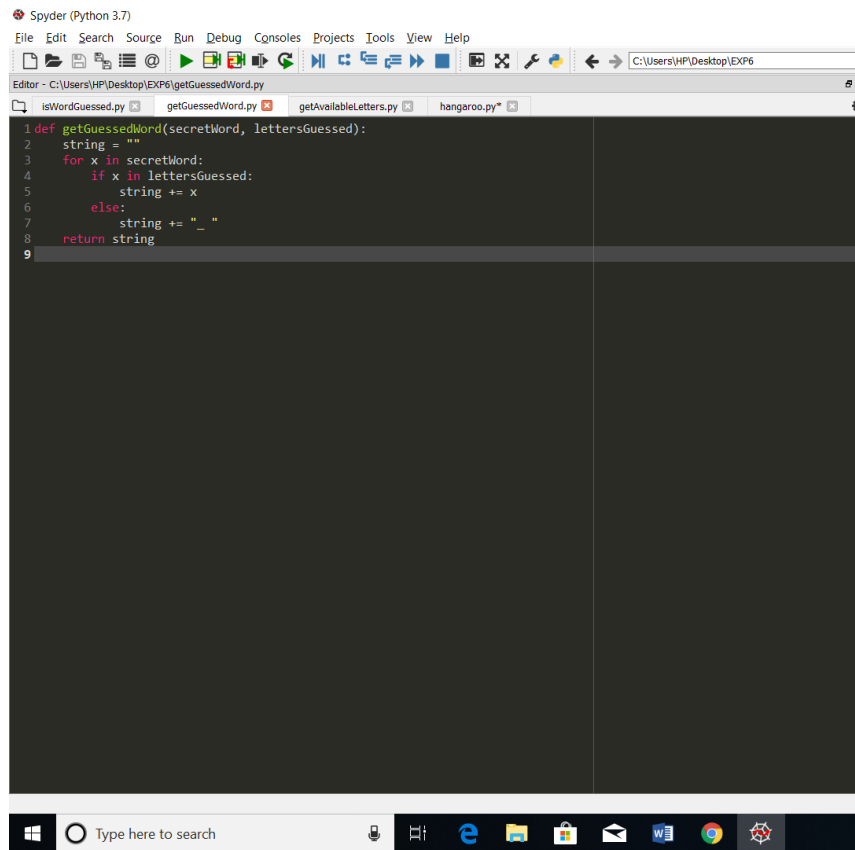


```
In [121]: secretWord = 'apple'
In [122]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [123]: print(isWordGuessed(secretWord, lettersGuessed))
False
In [124]:
```

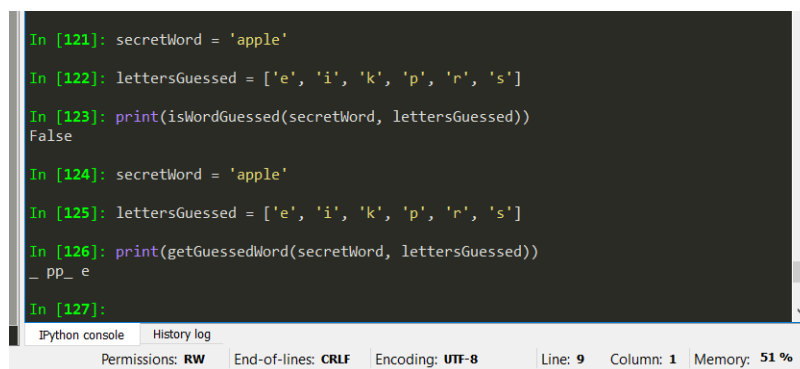
IPython console History log

Permissions: **RW** End-of-lines: **CRLF** Encoding: **UTF-8** Line: **5** Column: **20** Memory: **49 %**

2.)



```
1 def getGuessedWord(secretWord, lettersGuessed):
2     string = ""
3     for x in secretWord:
4         if x in lettersGuessed:
5             string += x
6         else:
7             string += "_ "
8     return string
9
```



```
In [121]: secretWord = 'apple'
In [122]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [123]: print(isWordGuessed(secretWord, lettersGuessed))
False
In [124]: secretWord = 'apple'
In [125]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [126]: print(getGuessedWord(secretWord, lettersGuessed))
_ pp_ e
In [127]:
```

IPython console History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 9 Column: 1 Memory: 51 %

3.)

The screenshot displays the Spyder Python IDE interface. The main editor window shows the following Python code:

```
1 def getAvailableLetters(lettersGuessed):
2     letters = "abcdefghijklmnopqrstuvwxyz"
3     string = ""
4     for x in letters:
5         if x not in lettersGuessed:
6             string += x
7     return string
```

The Variable explorer on the right shows the following variables:

Name	Type	Size	Value
guess	str	1	apple
guessesLeft	int	1	3
lettersGuessed	list	6	['e', 'i', 'k', 'p', 'r', 's']
secretWord	str	1	apple

The Python console on the bottom right shows the following execution output:

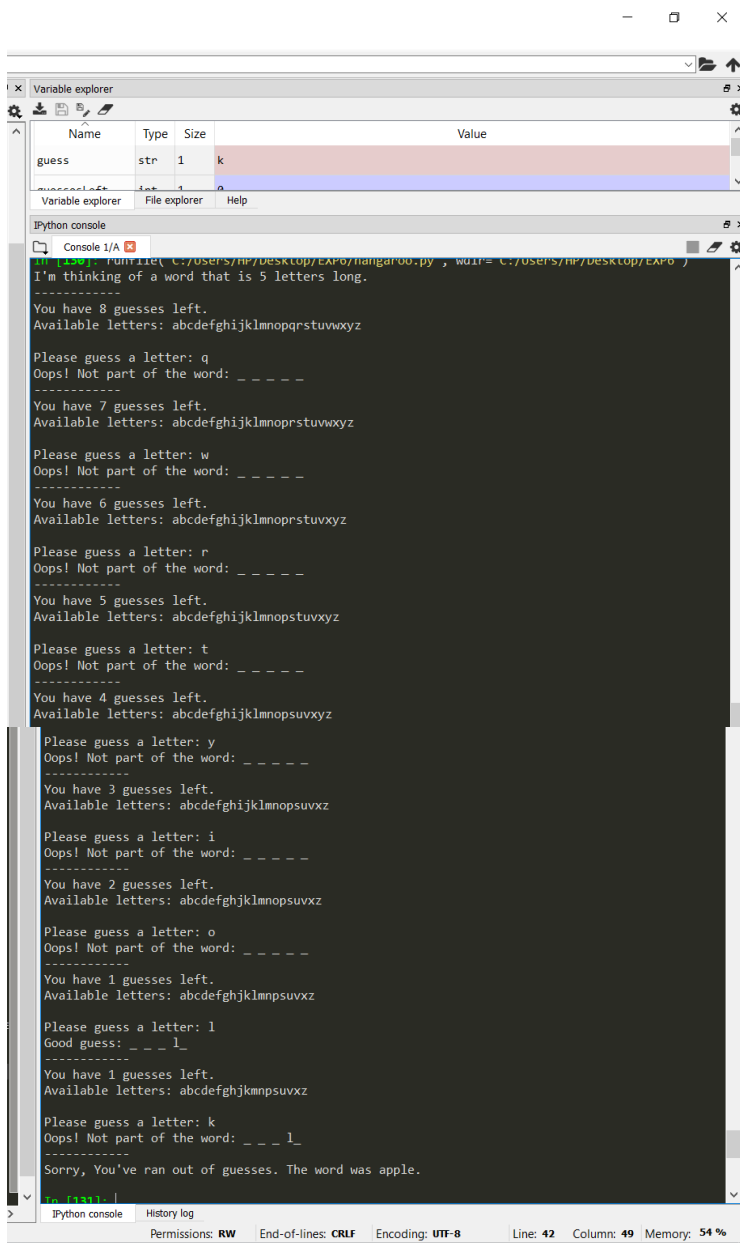
```
Good guess: apple
Congratulations! You've won!
In [121]: secretWord = 'apple'
In [122]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [123]: print(isWordGuessed(secretWord, lettersGuessed))
False
In [124]: secretWord = 'apple'
In [125]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [126]: print(getGuessedWord(secretWord, lettersGuessed))
_ PP_ e
In [127]: lettersGuessed = ['e', 'i', 'k', 'p', 'r', 's']
In [128]: print(getAvailableLetters(lettersGuessed))
abcdefghijklmnopqrstuvwxyz
In [129]:
```

The status bar at the bottom indicates: Permissions: RW, End-of-lines: CRLF, Encoding: UTF-8, Line: 7, Column: 18, Memory: 53 %.

4.)

```
Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\HP\Desktop\EXP6\hangaroo.py
isWordGuessed.py getGuessedWord.py getAvailableLetters.py hangaroo.py*
1 def isWordGuessed(secretWord, lettersGuessed):
2     for x in secretWord:
3         if x not in lettersGuessed:
4             return False
5
6 def getGuessedWord(secretWord, lettersGuessed):
7     word = ''
8     for x in secretWord:
9         if x in lettersGuessed:
10             word += x
11         else:
12             word += '_'
13     return word
14
15 def getAvailableLetters(lettersGuessed):
16     letters = "abcdefghijklmnopqrstuvwxyz"
17     string = ''
18     for x in letters:
19         if x not in lettersGuessed:
20             string += x
21     return string
22
23 secretWord = "apple"
24
25 def hangaroo(secretWord):
26     print("Welcome to the game, Hangaroo!")
27     print("I'm thinking of a word that is " + str(len(secretWord)) + " letters long.")
28     lettersGuessed = ''
29     guessesLeft = 8
30     print("-----")
31     while True:
32         print("You have " + str(guessesLeft) + " guesses left.")
33         print("Available letters: " + getAvailableLetters(lettersGuessed))
34         guess = input("Please guess a letter: ")
35         if guess in secretWord and guess not in lettersGuessed:
36             lettersGuessed += guess
37             print("Good guess: " + getGuessedWord(secretWord, lettersGuessed))
38         elif guess in lettersGuessed:
39             print("Oops! You've already guessed that letter: " + getGuessedWord(secretWord, lettersGuessed))
40         else:
41             lettersGuessed += guess
42         print("Oops! Not part of the word: " + getGuessedWord(secretWord, lettersGuessed))
43         guessesLeft -= 1
44         print("-----")
45         if guessesLeft <= 0:
46             print("Sorry, You've ran out of guesses. The word was " + secretWord + ".")
47             break
48         if guess == secretWord:
49             print("Congratulations! You've won!")
50             break
```

PROOF



The screenshot displays a Python IDE interface. At the top, a 'Variable explorer' window shows a table with columns 'Name', 'Type', 'Size', and 'Value'. It contains one entry: 'guess' of type 'str' with a size of 1 and a value of 'k'. Below this is a 'Python console' window. The console output shows a word-guessing game. The user has 8 guesses left and the word is 'apple'. The user has guessed 'q', 'w', 'r', 't', 'y', 'i', 'o', and 'l', all of which are incorrect. The game ends with the message 'Sorry, You've ran out of guesses. The word was apple.'

```
in [150]: runfile('C:/Users/nr/Desktop/EXP6/nalgar00.py', wdir='C:/Users/nr/Desktop/EXP6')
I'm thinking of a word that is 5 letters long.
-----
You have 8 guesses left.
Available letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: q
Oops! Not part of the word: _ _ _ _ _
-----
You have 7 guesses left.
Available letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: w
Oops! Not part of the word: _ _ _ _ _
-----
You have 6 guesses left.
Available letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: r
Oops! Not part of the word: _ _ _ _ _
-----
You have 5 guesses left.
Available letters: abcdefghijklmnopstuvwxyz

Please guess a letter: t
Oops! Not part of the word: _ _ _ _ _
-----
You have 4 guesses left.
Available letters: abcdefghijklmnopstuvxyz

Please guess a letter: y
Oops! Not part of the word: _ _ _ _ _
-----
You have 3 guesses left.
Available letters: abcdefghijklmnopstuvxz

Please guess a letter: i
Oops! Not part of the word: _ _ _ _ _
-----
You have 2 guesses left.
Available letters: abcdefghijklmnopstuvxz

Please guess a letter: o
Oops! Not part of the word: _ _ _ _ _
-----
You have 1 guesses left.
Available letters: abcdefghijklmnpsuvxz

Please guess a letter: l
Good guess: _ _ _ l _
-----
You have 1 guesses left.
Available letters: abcdefghjkmnpstuvxz

Please guess a letter: k
Oops! Not part of the word: _ _ _ l _
-----
Sorry, You've ran out of guesses. The word was apple.
```

Variable explorer

Name	Type	Size	Value
guess	str	1	apple
guessesleft	int	1	0

Variable explorerFile explorerHelp

IPython console

Console 1/A

I'm thinking of a word that is 5 letters long.

You have 8 guesses left.

Available letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: a

Good guess: a _ _ _ _

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: p

Good guess: app _ _

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: l

Good guess: appl _

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: e

Good guess: apple

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: apple

Good guess: apple

Congratulations! You've won!

IPython consoleHistory log

Permissions: RWEnd-of-lines: CRLFEncoding: UTF-8Line: 42Column: 49Memory: 54 %