Hangman Game (1)

Making Game with Python (1)

Zhihong (John) Zeng & Andrew Zeng

Today

- Review and test
- Check a letter in a string
- Hangman game flowchart
- choose_word
- is_word_guessed
- get_guessed_word
- guess_loop
- Hangman game

Review

- import time module
- Escape character and multiline string
- User-defined function
- While loop
- Boolean operators: and, or , not

Test

```
# find the bugs
A = ‘It’s a test’

# what will be printed
print(‘Welcome\nToday is Sunday’)

def addition(x, y=0):
        ans = x + y
        return ans

print(addition(1, 2))
print(addition(1))
```

Test

```
# find the bugs
A = ‘It’s a test’

# what will be printed
print(‘Welcome\nToday is Sunday’)

def addition(x, y=0):
        ans = x + y
        return ans

print(addition(1, 2))
print(addition(1))
```

```
# correction
A = ‘It\’s a test’
A = “It’s a test’

Welcome
Today is Sunday




3
1
```

Test

```
# what will be printed

x = 5
while x > 0:
        print(x)
        x -= 1




x = 5
while x:
        print(x)
        x -= 1
```

Test

```
# what will be printed

x = 5
while x > 0:
        print(x)
        x -= 1



x = 5
while x:
        print(x)
        x -= 1
```

0==False

```
# what will be printed

5
4
3
2
1

5
4
3
2
1
```

Check a item in a string

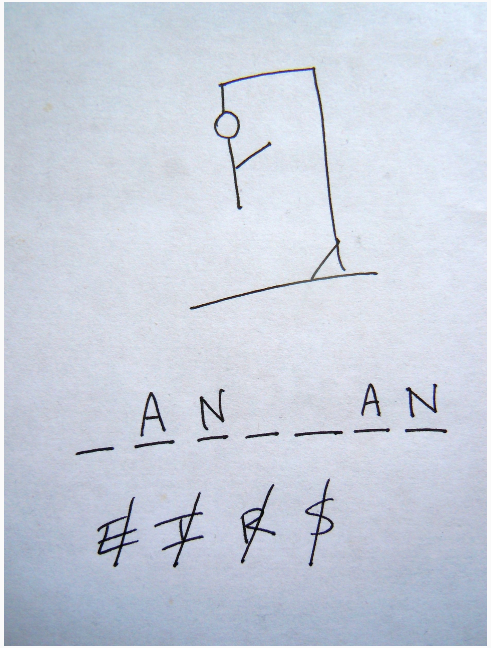
A = ‘abcd’

print(‘c’ in A) → True

print(‘e’ in A) → False

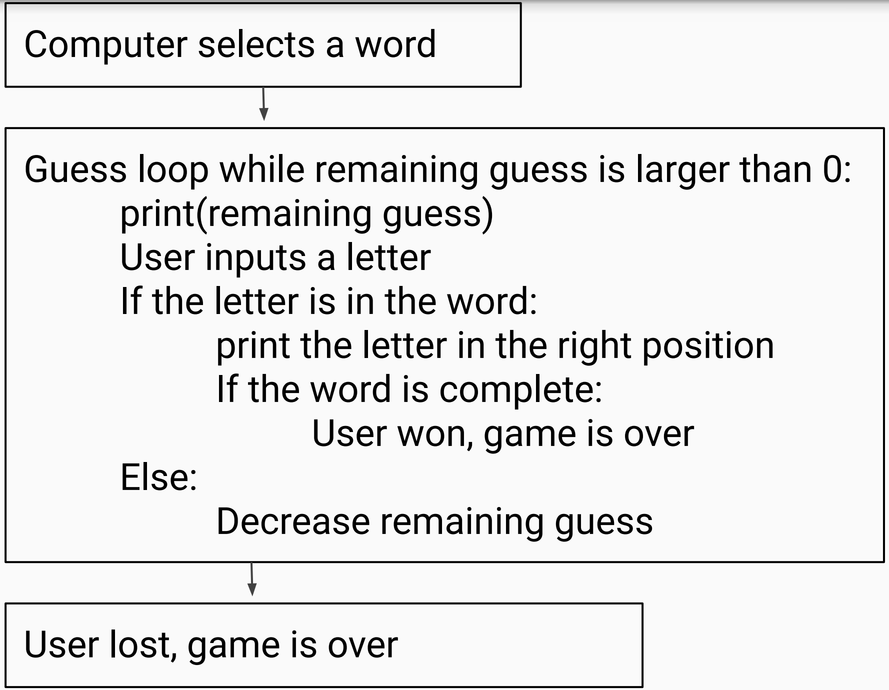
print(‘e’ not in A) → True

Hangman Game



demo

Hangman Game Flowchart



Computer selects a word

```
Guess loop while remaining guess is larger than 0:
    print(remaining guess)
    User inputs a letter
    If the letter is in the word:
        print the letter in the right position
        If the word is complete:
            User won, game is over
    Else:
        Decrease remaining guess
```

User lost, game is over

Choose word

```
import random
def choose_word():
    """choose word from a list
    Returns:
        string -- selected word
    """
    wordlist = 'ant bear cat dog beer'.split()
    w = random.choice(wordlist)
    return w

print(choose_word())
print(choose_word())
print(choose_word())
```

is_word_guessed

```
def is_word_guessed(secrete_word, letters_guessed):
    """check whether all of letters of secrete word have been guessed
    Arguments:
        secrete_word {string} -- secrete word
        letters_guessed {string} -- guessed letters
    Returns:
        Boolean -- True if all letters of the word are found in the letter_guessed.
                    Otherwise False.
    """
    for x in secrete_word:
        if x not in letters_guessed:
            return False
    return True

print(is_word_guessed('banana', 'abn'))
print(is_word_guessed('bead', 'earb'))
```

get_guessed_word

```
def get_guessed_word(secrete_word, letter_guessed):
    """Get the word with guessed letters
    Arguments:
        secrete_word {string} -- secrete word
        letter_guessed {string} -- guessed letters
    Returns:
        string -- word with guessed letters
    """
    word = ''
    for x in secrete_word:
        if x in letter_guessed:
            word += x
        else:
            word += '_'
    return word

print(get_guessed_word('hangman', 'hamg'))
```

Guess loop

```
def guess_loop(secrete_word, max_guess):
    remaining_guess = max_guess
    guessed = ''
    while remaining_guess > 0:
        print('You have {} guesses left'.format(remaining_guess))
        letter = input('Please guess a letter: ')
        letter = letter.lower()
        if letter in secrete_word:
            guessed += letter
            print('Good guess: {}'.format(get_guessed_word(secrete_word, guessed)))
            if is_word_guessed(secrete_word, guessed):
                print('Congratulations, You won!\n')
                return
        else:
            print('Oops! That letter is not in my word: {}'.format(get_guessed_word(secrete_word, guessed)))
            remaining_guess -= 1

    print('Sorry, you ran out of guesses. The word was {}\n'.format(secrete_word))

guess_loop('bear', 3)
```

Hangman game

```
def hangman(max_guess):
    secrete_word = choose_word()

    print('''Welcome to the game Hangman!
    I am thinking of a word that is {} letters long.
    '''.format(len(secrete_word)))

    guess_loop(secrete_word, max_guess)

hangman(4)
```