import re

import random

# Get the answer.

pool\_file = open("hangman-sample-answer-pool.txt")

pool\_answers = []

pool\_answer\_line = pool\_file.readline()

while pool\_answer\_line:

pool\_answers.append(pool\_answer\_line)

pool\_answer\_line = pool\_file.readline()

pool\_file.close()

answer = random.choice(pool\_answers)

answer = answer.upper()

# pre game setup.

answer\_guessed = []

for current\_answer\_character in answer:

if re.search("^[A-Z]$", current\_answer\_character):

answer\_guessed.append(False)

else:

answer\_guessed.append(True)

# Game logic.

num\_of\_incorrect\_guesses = 5

current\_incorrect\_guesses = 0

letters\_guessed = []

# User gameplay logic

while current\_incorrect\_guesses < num\_of\_incorrect\_guesses and False in answer\_guessed:

# Display game summary

print(f"Number of incorrect guesses remaining: {num\_of\_incorrect\_guesses - current\_incorrect\_guesses}")

print("Guessed Letters: ", end="")

for current\_letter\_guessed in letters\_guessed:

print(current\_letter\_guessed, end=" ")

print()

#Display puzzle board.

for current\_answer\_index in range(len(answer)):

if answer\_guessed[current\_answer\_index]:

print(answer[current\_answer\_index], end="")

else:

print("\_",end="")

print()

#let user guess a letter

letter = input("Enter a letter: ")

letter = letter.upper()

#check if user entered a valid letter

if re.search("^[A-Z]$", letter) and len(letter) == 1 and letter not in letters\_guessed:

#insert the letter guessed by user (insertionnn sort)

current\_letter\_index = 0

for current\_letter\_guessed in letters\_guessed:

if letter < current\_letter\_guessed:

break

current\_letter\_index += 1

letters\_guessed.insert(current\_letter\_index, letter)

#check if letter is in the puzzle

if letter in answer:

for current\_answer\_index in range(len(answer)):

if letter == answer[current\_answer\_index]:

answer\_guessed[current\_answer\_index] = True

else:

current\_incorrect\_guesses += 1

#post game summary

if current\_incorrect\_guesses < num\_of\_incorrect\_guesses:

print("Congrats, you won!")

else:

print(f"Sorry, you lost. The answer was {answer}")