assingment2-1

September 16, 2024

[13]:

```python
# Create a function that would accept two inputs: a sentence(string),
# and a list containing bad words that the user would like to censor but not remove.
# The function should return the newly filtered sentence wherein
# the bad words are replaced with asterisks equal to the length of the censored word.

# Task 1: Censoring Bad Words in a Sentence
def censor_sentence(sentence: str, bad_words: list) -> str: # this is a return type annotation, this indicate that the retun value is string
    for bad_word in bad_words: # the loop starts
        sentence = sentence.replace(bad_word, '*' * len(bad_word))
    return sentence

# Predefined list of bad words to censor
bad_words = ["bastard", "bitch", "rude", "horrible", "demon", "go to hell", "dickhead"]

# Get user input for the sentence
sentence = input("Enter a sentence: ")

# Call the function with predefined bad words list and print the censored sentence
censored_sentence = censor_sentence(sentence, bad_words)
print("Censored Sentence:", censored_sentence)

# only the bad_word define list will be censored. If you supply anything that out of the list will show in the sentence.
```

```
Censored Sentence: This is a Bad example of a ******** sentence.
```

[21]:

```python
# Create a quadratic equation solver module that would write the inputs of
# the user and the corresponding output into text files.

import math

# Function to solve the quadratic equation
```

```python
def solve_quadratic(a: float, b: float, c: float) -> tuple: #will return tuple type
    discriminant = b**2 - 4*a*c  # Calculate the discriminant
    if discriminant > 0:
        root1 = (-b + math.sqrt(discriminant)) / (2*a)
        root2 = (-b - math.sqrt(discriminant)) / (2*a)
        return root1, root2
    elif discriminant == 0:
        root = -b / (2*a)
        return root, root  # One real root (double root)
    else:
        return None, None  # No real roots (complex roots)

# Function to write inputs and results to a text file
# found in this folder
def write_to_file(data: str):
    with open('quadratic_solver_results.txt', 'a') as file:
        file.write(data + '\n')

# Main function to take user input, solve the equation, and write the results
def quadratic_solver():
    # Get user input for coefficients a, b, and c
    a = float(input("Enter coefficient a: "))
    b = float(input("Enter coefficient b: "))
    c = float(input("Enter constant c: "))

    # Solve the quadratic equation, looping to the def solve_quadratic line
    roots = solve_quadratic(a, b, c)

    # Prepare the output for writing to the file
    equation = f"Quadratic Equation: {a}x^2 + {b}x + {c} = 0"

    if roots == (None, None):
        result = "No real roots."
    elif roots[0] == roots[1]:
        result = f"One real root: x = {roots[0]}"
    else:
        result = f"Two real roots: x1 = {roots[0]}, x2 = {roots[1]}"

    # Write the inputs and the result to a file
    write_to_file(equation)
    write_to_file(result)

    # Display the result to the user
    print("Results:")
    print(result)
```

```python
# Run the quadratic solver
quadratic_solver()
```

```
Results:
No real roots.
```

[ ]:

```
```