**Name:** Jadyn Brabham  
**Date Created:** 02/03/2025

**Program Description:**This program will create a list of 30 words and phrases commonly found in spam messages. The program will ask the user to input an email message, and it will scan the message for each of those 30 words or phrases. For each occurrence of those words and phrases in the message, a point will be added to the message’s “spam score”. Then, based on the “spam score” it will rate the likelihood that the message is spam. Last, the program will display the user’s “spam score”, likelihood that the message is spam, and the words/phrases which caused it to be spam.

**Functions used in the Program (list in order as they are called):**1. **Function Name:** main()  
**Description:** The main function defines the spam keywords/phrases, asks the user to input an email message for the program to scan for spam keywords/phrases, and then uses the calculate\_spam\_score() and determine\_spam\_likelihood() functions to analyze the email message. Then it displays the spam score, likelihood of the message being spam, and the keywords/phrases that were found.

**Parameters:** None  
**Variables:**

1. spam\_keywords: a list of 30 common spam keywords/phrases
2. email\_message: holds the email message entered by the user
3. spam\_score: the total number of keywords/phrases found in the email message
4. found\_keywords: list of keywords/phrases that were found in the email message
5. spam\_likelihood: a string indicating the likelihood that the message is spam

**Logical Steps:**

1. Define a list of 30 common spam keywords/phrases.
2. Prompt the user for the input to receive the email message.
3. Pass the email\_message and spam\_keywords to the calculate\_spam\_score function to calculate the spam score and find the matching keywords/phrases.
4. Call the spam\_likelihood function to determine the likelihood of the message being spam based on the spam score.
5. Display the spam score, likelihood of the message being spam, and the list of keywords/phrases found in the message.

**Returns:** The function does not return anything, instead it prints the output to the user.

2. **Function Name:** calculate\_spam\_score()

**Description:** The calculate\_spam\_score() function scans the user’s email message for the presence of spam keywords/phrases and calculates the total spam score. For each occurrence of a keyword/phrase, the spam score is incremented, and the keyword/phrase is added to the list of found keywords.  
**Parameters:**

1. email\_message: holds the email message entered by the user
2. spam\_keywords: a list of 30 common spam keywords/phrases

**Variables:**

1. spam\_score: integer that tracks the number of keyword/phrases found in the email message
2. found\_keywords: list that stores the keywords/phrases found in the email message

**Logical Steps**:

1. Initialize the variable spam\_score to 0 and found\_keywords as an empty list
2. Loop through each keyword/phrase in the spam\_keywords list
3. Convert both the keyword and email\_message to lowercase and for each keyword, check if it exists in email\_message
4. If a match is found, increase the spam\_score by 1 and add the keyword/phrase to the found\_keywords list
5. Return the spam\_score and the found\_keywords list

**Returns:**

1. spam\_score: integer representing the total number of keywords/phrases found in the email message
2. found\_keywords: list of keywords/phrases found in the email message

3.**Function Name:** determine\_spam\_likelihood()

**Description:** The determine\_spam\_likelihood() function assesses the likelihood of an email message being spam based on the calculated spam score. It returns a string indicating the level of spam likelihood: high, medium, or low.

**Parameters:**

1. spam\_score: integer representing the total number of keywords/phrases found in the email message

**Variables:**

1. spam\_score: integer representing the total number of keywords/phrases found in the email message

**Logical Steps:**

1. If spam\_score is greater than or equal to 10, return “High likelihood the message is spam.”
2. If spam\_score is between 5 and 9, return “Medium likelihood the message is spam.”
3. If spam\_score is less than 5, return “Low likelihood the message is spam.”

**Returns:**

1. spam\_likelihood: string that describes the likelihood of the email message being spam based on the spam score

**Logical Steps:**

1. Call the main function
2. The calculate\_spam\_score function and the determine\_spam\_likelihood function is called in the main function

**Link to your repository:** https://github.com/jbrabham23/COP2373/tree/master