File Handling Problems

1. **Create and Write to a File**
   Write a program to create a file named data.txt and write the string "Hello, World!" into it.
2. **Read from a File**
   Open data.txt and display its contents on the screen.
3. **Append to a File**
   Add the text "Welcome to Python programming!" to data.txt.
4. **Count Lines in a File**
   Count the number of lines in data.txt.
5. **Count Words in a File**
   Write a program to count the number of words in data.txt.
6. **Copy File Contents**
   Create a program that copies the contents of data.txt to a new file called copy.txt.
7. **Check if File Exists**
   Check if a file named data.txt exists in the current directory.
8. **Read File Line by Line**
   Read data.txt line by line and print each line.
9. **Search for a Word in a File**
   Write a program to search for the word "Python" in data.txt and print the line(s) where it appears.
10. **Write a List to a File**
    Write a list of numbers [1, 2, 3, 4, 5] to a file named numbers.txt.
11. **Reverse File Contents**
    Read data.txt and write its contents in reverse order to a file named reverse.txt.
12. **File Statistics**
    Write a program to display the number of characters, words, and lines in data.txt.
13. **Merge Two Files**
    Merge the contents of data.txt and numbers.txt into a file named merged.txt.

14. **Count Occurrences of a Word**
    Count how many times the word "Python" appears in data.txt.
15. **Remove a Word from a File**
    Write a program to remove the word "Hello" from data.txt.
16. **File Encryption**
    Encrypt the contents of data.txt by shifting each character by 2 positions (Caesar cipher) and save it to encrypted.txt.
17. **File Decryption**
    Decrypt the file encrypted.txt to retrieve the original content.
18. **Remove Blank Lines**
    Remove all blank lines from data.txt.
19. **Find Longest Word in a File**
    Write a program to find the longest word in data.txt.
20. **Word Frequency Analysis**
    Count the frequency of each word in data.txt and display the results.

Error Handling Problems

21. **Handle File Not Found Error**
    Write a program to handle the case when data.txt does not exist while reading it.
22. **Handle Division by Zero**
    Write a program that handles a ZeroDivisionError when dividing two numbers.
23. **Invalid Input Handling**
    Handle the case when the user enters a non-integer value for integer input.
24. **Handle Key Error**
    Write a program to handle a KeyError when accessing a dictionary.
25. **File Read Permission**
    Handle the case when the program does not have permission to read data.txt.
26. **Catch Multiple Exceptions**
    Write a program that handles both FileNotFoundError and PermissionError.

27. **Custom Exception**

    Create a custom exception called NegativeNumberError and raise it if a user inputs a negative number.

28. **Handle IndexError**

    Handle the case when accessing an index that does not exist in a list.

29. **Nested Exception Handling**

    Write a program with a try...except block inside another try...except block for different types of errors.

30. **Resource Cleanup with finally**

    Write a program to read a file and ensure it is always closed, even if an error occurs.