Column Permutation Cryptanalysis – Programming Directions

**Goal:** Write a VB program, which will aid in the cryptanalysis of the column permutation cipher.

- Create a form to handle input and output.
- **Specific calculations**
  - List out all possible rectangles (factors) for the ciphertext length. *This will help the user determine the possible keyword lengths.*
    - Example: There are 102 characters in the ciphertext
    - Possible rectangles: 1 x 102, 102 x 1, 2 x 51, 51 x 2, 3 x 34, 34 x 3, 6 x 17, 17 x 6
    - *Consider the largest number that you have to use in relation to the length.*
  - Calculated vowel differences for each rectangle. *This should help the user determine the correct keyword length.*
    - We expect 40% of a row to be vowels.
    - Calculate the absolute difference between the actual number of vowels and the expected number of vowels.
    - Find the total absolute differences for all rows within a rectangle.
    - Output a list of dimensions with the accompanying sums.
  - Calculate the centiban counts. *This will help the user determine the order of the columns.*
    - When the user selects a possible rectangle centiban counts are calculated for all possible combinations of columns.
    - Example
      - If there are four columns then the following columns need to be compared (1,2), (1,3), (1,4), (2,1), (2,3), (2,4), (3,1), (3,2), (3,4), (4,1), (4,2), (4,3).
      - For each combination the centiban weight for each diagram needs to be looked up and then a sum for that column needs to be calculated.
      - Output a list of column pairs with the accompanying centiban sums.
    - A table is needed where the user can move the columns around in order to find the correct order for the columns.
    - Centiban sums should still be visible to aid in discussion making.
    - Column numbers should move with the columns.
- **Other things to include**
  - Output the plaintext from the table to a textbox.
  - Appropriate use of tab in the code (organization)
  - Appropriate documentation (comments) in the code. *This will be very important as this program will get complex quickly.*
  - A function for removing all formatting from the input text.
  - Some features that check for errors
- **Extra features**
  - Ability to open text files for the input.
  - Ability to pull the centiban values from a database.
  - An auto solver.
**Grading:**

<table>
<thead>
<tr>
<th>Features</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly prompts and handles inputs</td>
<td>/4</td>
</tr>
<tr>
<td>Properly calculates all possible rectangles and displays them</td>
<td>/8</td>
</tr>
<tr>
<td>Properly calculates the absolute vowel difference for each possible rectangle</td>
<td>/10</td>
</tr>
<tr>
<td>Properly calculates centiban totals for each column combination</td>
<td>/12</td>
</tr>
<tr>
<td>Table that easily allows the user to reorder the columns</td>
<td>/8</td>
</tr>
<tr>
<td>Outputs the plaintext from the table</td>
<td>/2</td>
</tr>
<tr>
<td>Includes some error checking</td>
<td>/4</td>
</tr>
<tr>
<td>Function and implementation of the function for removing formatting</td>
<td>/2</td>
</tr>
<tr>
<td>Appropriate use of tab (organization)</td>
<td>/1</td>
</tr>
<tr>
<td>Appropriate use of documentation</td>
<td>/4</td>
</tr>
<tr>
<td>Properly turns in all associated files (and compiled)</td>
<td>/1</td>
</tr>
<tr>
<td>Total</td>
<td>/56</td>
</tr>
</tbody>
</table>