Algorithm 1: The confusion function algorithm

Input: $S = [W_1, W_2, ..., W_n]$, single sub-scripted text array of maximum size 100 words, matrix to be inputted.
N = non-negative value $\leq \hat{S}$ items number, refers to the number of words that have to be changed

Output: $\hat{S} = [\hat{W}_1, \hat{W}_2, ..., \hat{W}_n]$, single sub-scripted text array of maximum size 100 words, matrix to be generated.

initialization $\hat{S} = S$;
for N **do**
  $Error - type1$ = 40% of N
  for $Error - type1$ **do**
    Select a random word in $\hat{S}$ = W_i
    Add W_i in the index of W_{i+1}
    Update $\hat{S}$
  end
  $Error - type2$ = 40% of N
  for $Error - type2$ **do**
    Select a random word in $\hat{S}$ = W_i
    Delete W_i
    Update $\hat{S}$
  end
  $Error - type2$ = 10% of N
  for $Error - type3$ **do**
    Select a random word in $\hat{S}$ = W_i
    Decomposing word W_i to an array of characters = $[C_1, C_2, ..., C_n]$
    for $W_i = [C_1, C_2, ..., C_n]$ **do**
      Select a random character in $[C_1, C_2, ..., C_n]$ = C_i
      Add C_i in the index of C_{i+1}
      Composing $[C_1, C_2, ..., C_n]$ to $\hat{W}_i$
    end
    Overwrite $\hat{W}_i$ with in $\hat{S}$ in the index of W_i
    Update $\hat{S}$
  end
  $Error - type2$ = 10% of N
  for $Error - type4$ **do**
    Select a random word in $\hat{S}$ = W_i
    Decomposing word W_i to an array of characters = $[C_1, C_2, ..., C_n]$
    for $W_i = [C_1, C_2, ..., C_n]$ **do**
      Select a random character in $[C_1, C_2, ..., C_n]$ = C_i
      Delete C_i
      Composing $[C_1, C_2, ..., C_n]$ to $\hat{W}_i$
    end
    Overwrite $\hat{W}_i$ with in $\hat{S}$ in the index of W_i
    Update $\hat{S}$
  end
end