Podczas laboratorium nr 03 kontynuujemy pracę z algorytmami dotyczącymi sortowania i do tego dodamy modyfikację danych wejściowych z liczb na ciągi znaków. Indywidualnie wybieracie metodę porównania dwóch ciągów znaków (np. 3.0 poprzez długość, 4.0 poprzez sumę kodów ASCII znaków w ciągu, 5.0 poprzez kolejność alfabetyczną).

Implementujemy znany wcześniej algorytm oraz jeden z algorytmów opisanych w wykładzie 03. Pamiętajcie, że macie to zrealizować algorytmicznie (korzystając z pętli, instrukcji sterujących, etc), a nie „językowo" (użycie sorter() czy find()).

1. **Algorytm znajdywania Palindromów - prosty.**
   Algorytm musi sprawdzić ile ciągów znaków w pliku to palindromy i wypisać je do drugiego pliku.

2. **Algorytm znajdywania Anagramów - prosty.**
   Algorytm musi sprawdzić ile par ciągów znaków w pliku to anagramy i wypisać je (wraz z ich indeksami w pliku) do drugiego pliku.

3. **Szyfrowanie poprzez odbicie lustrzane.**
   Algorytm powinien wczytać tekst „zadany" i poprzez odbicie lustrzane zakodować tekst, a następnie (za pomocą tej samej funkcji) zdekodować.

4. **Szyfr Cezara (kodowanie) - prosty.**
   Korzystając z materiałów do wykładu oraz własnej inwencji, zaprojektujcie algorytm szyfrowania oparty o tzw. szyfr Cezara. Algorytm powinien przyjmować (na razie pobiera od użytkownika, nie z pliku) ciąg znaków zwany dalej „zadanym" i liczbę z przedziału od 1 do 25 (mamy 26 wielkich i 26 małych liter). Następnie, zgodnie z szyfrem przestawieniowym, kodować „zadany" tekst do zmiennej „zakodowany" i wyświetlać go na ekranie. W tej wersji nie będzie istotne aby zakodowana litera nadal była literą, czyli nie musicie dbać o to, żeby przy kluczu o wartości 3 z litery Z powstała litera C.

   Należy również przygotować funkcję dekodującą, która zaszyfrowany tekst przekształci na tekst oryginalny („zadany").

5. **Algorytm Brute Force – wyszukiwanie ciągu znaków w innym ciągu znaków.**
   Korzystając z materiałów z wykładu zaimplementujcie algorytm, który wskaże czy „szukany” ciąg znaków występuje w „przeglądanym”. Dla przykładu:
   szukany: AAB
   przeglądany: BAABABAAABAAB
   Odpowiedź: tak, występuje (pierwszy wystąpienie pod indeksem 1)