Laboratorium nr 3: Algorytm LSB dla plików BMP – część 2

Zalecany język programowania: C#

## Zadanie:

Program ma ukrywać informacje w następujący sposób:

<u>Dane do ukrycia</u> -> szyfrowanie ->kodowanie nadmiarowe->permutacja rozpraszająca-> kodowanie za pomocą parzystości -> zakodowanie w subpikselach -> permutacja odwrotna do rozpraszającej -><u>Dane ukryte</u>

W związku z tym należy rozbudować program z poprzednich zajęć o następujące funkcje:

## • Kodowanie nadmiarowe:

 Dane po zaszyfrowaniu mają zostać zakodowane z użyciem kodu nadmiarowego. Po zaszyfrowaniu, dane kodujemy kodem nadmiarowym.
 Kod nadmiarowy ma mieć postać kodu 1 z 5. Czyli 1 bit kodujemy za pomocą 5 bitów z możliwością poprawy dwóch bitów.

## • Klucz steganograficzny:

Aby ukrywany tekst nie były umieszczany na końcu wiadomości lub w dowolnym innym stałym miejscu należy użyć klucz steganograficzny do rozproszenia ukrywanego tekstu na cały plik BMP za pomocą permutacji rozpraszającej i odwrotnej opisanych w następnych podpunktach. Kluczem jest tablica bajtów tworzona z hasła za pomocą funkcji SHA512.

## • Permutacja rozpraszająca (oraz permutacja odwrotna):

- Aby nie umieszczać informacji na początku pliku BMP i rozproszyć informację po całym pliku wykonujemy permutację macierzy z pikselami i następnie na początku macierzy po permutacji umieszczamy ukrywaną informację. Permutacja odwrotna spowoduje przywrócenie oryginalnego układu pikseli i tym samy rozproszenie ukrywanej informacji na cały plik.
- Permutacje można wykonać na jeden z dwóch sposobów zaprezentowanych poniżej (proszę zwrócić uwagę, że macierz dla zdjęcia 4000x3000 pikseli będzie miała 12 milionów elementów, co wpłynie znacząco na wydajność algorytmu):
  - Z klucza wywodzimy jednoznacznie numer i-tej permutacji (wg porządku leksykograficznego) i używamy odpowiedniego algorytmu do jej obliczenia, przykład patrz: <a href="http://stackoverflow.com/questions/7918806/finding-n-th-permutation-without-computing-others">http://stackoverflow.com/questions/7918806/finding-n-th-permutation-without-computing-others</a>;
  - Klucz steganograficzny będzie ziarnem dla generatora liczb losowych.
    Losujemy liczby losowe, które będą oznaczały kolejne elementy
    macierzy i w ten sposób tworzymy macierz permutacji. Przy
    losowaniu należy pamiętać, aby sprawdzić, czy element nie został
    wcześniej wylosowany.