**Wybrane algorytmy kryptograficzne**

***Szyfrowanie symetryczne*** [***http://szyfrowanie.estrefa.net/***](http://szyfrowanie.estrefa.net/)

**Szyfrowanie symetryczne** (szyfr z jednym/pojedynczym kluczem) – nadawca i odbiorca wiadomości używają tego samego **prywatnego klucza**. Inaczej mówiąc, każdy kto jest w posiadaniu tego klucza symetrycznego, może zarówno zaszyfrować jak i odszyfrować wiadomość. Wobec tego oczywiste jest, że szyfrowanie symetryczne będzie skuteczne tylko wówczas jeżeli ten klucz nie wpadnie w niepowołane ręce – kluczem tym powinny dysponować tylko nadawca i odbiorca.

1. Nadawca przekazuje odbiorcy (lub odwrotnie) klucz prywatny przez bezpieczny kanał.
2. Nadawca szyfruje wiadomość za pomocą klucza prywatnego. Szyfrowania odbywa się za pomocą specjalnego **algorytmy (szyfru) symetrycznego** który używa klucza prywatnego. Do szyfrów symetrycznych zaliczamy takie algorytmy jak DES, AES, Blowfish, Triple DES, Twofish, Serpent, RC4, RC5. Zaszyfrowana wiadomość zostaje wysłana to odbiorcy.
3. Odbiorca przy pomocy klucza prywatnego odszyfrowuje wiadomość.

Z uwagi na opisane powyżej niedogodności związane z przekazywaniem klucza prywatnego (patrz pkt 1), często zamiast szyfrowania symetrycznego wykorzystane jest [**szyfrowanie asymetryczne**](http://szyfrowanie.estrefa.net/Szyfrowanie-asymetryczne).

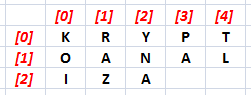
**Algorytmy przestawieniowe**

Przestawianie liter w tekście wiadomości jest jednym z najprostszych sposobów utajniania informacji. Zmiany kolejności znaków dokonuje się zwykle stosując pomocniczą tablicę najczęściej dwuwymiarową. Szyfrowanie przebiega w dwóch etapach:

1. Tekst jawny wpisuje się w określony sposób do tablicy, co nazywamy ścieżką zapisu.
2. Szyfr otrzymuje się przez odczyt tekstu z tablicy w określony sposób, co nazywamy ścieżka odczytu.

**Przestawienie kolumnowe** – ta metoda wykorzystuje tablicę dwuwymiarową. Tekst jawny jest wprowadzany znak po znaku (bez spacji) do kolejnych wierszy tablicy. **Kluczem** jest tu dowolny ciąg (permutacja) numerów kolumn. Szyfr powstaje przez odczyt znaków kolumnami zgodnie z kluczem.

***Przykład*** Zaszyfrujmy słowo KRYPTOANALIZA. Zastosujemy tablicę dwuwymiarową o 5-ciu kolumnach. Kluczem będzie ciąg numerów kolumn np. taki: 2, 1, 4, 0, 3. Po wpisaniu słowa do tablicy dostajemy:

Po odczytaniu dostajemy szyfr: YNARAZTLKOIPA.

***Ćwiczenie 1*** Napisz program zawierający funkcję **szyfruj**, która otrzymuje poprzez parametry: tekst jawny (bez spacji), ilość kolumn w tablicy oraz tablicę jednowymiarową z ciągiem numerów kolumn (czyli klucz). Funkcja zwraca zaszyfrowany tekst.

***string szyfruj(string tekst, int klucz[], int n)***

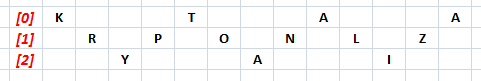
Jak będzie wyglądał zaszyfrowany tekst (bez spacji): UCZCIWOSC ZBIERA POCHWALY I UMIERA Z ZIMNA, w tablicy o 4-ch kolumnach z kluczem: 3, 2, 0, 1.

***Ćwiczenie 2*** Napisz program zawierający funkcję **deszyfruj**, która otrzymuje poprzez parametry: tekst szyfru (bez spacji), ilość kolumn w tablicy oraz tablicę jednowymiarową z ciągiem numerów kolumn (czyli klucz). Możesz założyć, że długość szyfru będzie podzielna przez ilość kolumn (czyli wszystkie wiersze w tablicy pomocniczej są „pełne”). Funkcja zwraca tekst jawny (bez spacji).

***string deszyfruj(string tekst, int klucz[], int n)***

Odszyfruj: AWYKWIAPCSSOYERAZTZCZ, gdy ilość kolumn=3, a klucz to: 2, 0, 1.

**Szyfr płotowy** – w metodzie tej tekst jawny zapisuje się tak, aby znaki tworzyły kształt płotu zbudowanego ze sztachet. Szyfr uzyskujemy, odczytując znaki „z płotu” wierszami. **Klucz** tego algorytmu to wysokość płotu, czyli liczba wierszy. Przy szyfrowaniu pomijamy spacje.

***Przykład*** Zaszyfrujmy słowo KRYPTOANALIZA. Zastosujemy płot o wysokości 3-ech wierszy.

Odczytując szyfr wierszami mamy: KTAARPONLZYAI

***Ćwiczenie 3*** Napisz program zawierający funkcję **szyfruj**, która zaszyfruje metodą płotową tekst: WLASNE MALE OGNISKO CENNIEJSZE OD STOSU ZLOTA. Zastosuj płot wysokości 2. Spacje pomijamy.

***string szyfruj(string tekst)***

***Ćwiczenie 4*** Napisz program zawierający funkcję **deszyfruj**, która odszyfruje tekst: BZOIJPYECYNSNEAEDOZNUZNCDOCK zaszyfrowany metodą płotową. Zastosowano płot wysokości 3 i pominięto spacje.

***string deszyfruj(string tekst)***

**Algorytmy podstawieniowe**

Szyfrowanie takie polega na zastępowaniu znaków tekstu jawnego innymi znakami (jednym lub kilkoma). Na dodatek można korzystać z jednego lub kilku alfabetów szyfrowych.

**Szyfry monoalfabetyczne** – zmieniają **każdy** znak tekstu jawnego na **dokładnie jeden** znak szyfru. Klasycznym przykładem jest tu **szyfr Cezara**.

***Przykład – szyfr Cezara***

Zasada szyfrowania i deszyfrowania tego szyfru jest taka sama. Do szyfrowania używamy alfabetu

**A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**

Daną literę zamieniamy na literę, która jest od niej o **k** pozycji dalej w alfabecie. Jeśli brakuje liter, wracamy do początku. Liczbę pozycji **k** nazywamy **kluczem.** Klucz oznaczamy liczbą od 0 do 26 (jeśli k=0 lub k=26 to szyfr jest identyczny z tekstem jawnym). Alfabet po przesunięciu z kluczem=3:

**D E F G H I J K L M N O P Q R S T U V W X Y Z A B C**

Taka konwencja jest dość wygodna, zwłaszcza, kiedy chcemy przejść do bardziej skomplikowanych szyfrów Vigenere'a. Jeśli chcemy zaszyfrować tekst KOD LEONARDA kluczem **3**, zamieniamy każdą literę na trzecią po niej, tj. **k** przechodzi na **n**, **o** na **r**, itd. W rezultacie otrzymujemy NRG OHRQDUGD. Rozszyfrowujemy według tej samej zasady, przy czym kluczem jest liczba **26-k**, gdzie **k** jest kluczem, którym szyfrowany był tekst. Na przykład, jeśli tekst BEQMZLHI AHGNZWE był zaszyfrowany kluczem **8**, to kluczem rozszyfrowującym jest **18**. Stosując ten klucz otrzymujemy tekst jawny TWIERDZA SZYFROW.

***Ćwiczenie 5*** Napisz program zawierający funkcję **szyfruj**, która zaszyfruje tekst jawny z kluczem k.

***string szyfruj(string tekst, int k)***

Nie musisz pisać funkcji **odszyfruj** w celu zobaczenia tekstu jawnego. Wystarczy ponownie wywołać funkcję **szyfruj** z kluczem 26-k.

Szyfr Cezara jest bardzo łatwy do złamania gdyż alfabet szyfrowy zawiera kolejne litery (więc maksymalnie 26 prób wystarczy do złamania tego szyfru). Aby znacznie utrudnić łamanie wprowadza się różne kolejności (permutacje) alfabetu. Popularna jest tu metoda polegająca na zastosowaniu **słowa lub zadania kluczowego**. Najpierw ze słowa lub zdania usuwa się wszystkie spacje i powtarzające się litery. Następnie ustawia się je na początku alfabetu szyfrowego. Po ostatniej literze dopisuje się pozostałe litery w kolejności alfabetycznej. Tak zbudowany alfabet szyfrowy jest dużo trudniejszy do złamania z powodu ogromnej liczby słów lub zdań kluczowych. Znane są przypadki stosowania jako słów lub zdań kluczowych umówionych fragmentów książek, do których dostęp ma nadawca jak i odbiorca utajnionej informacji.

***Przykład*** Przyjmijmy, że zdaniem kluczowym jest: KRYPTOGRAFIA I KRYPTOANALIZA. Tekstem jawnym, który chcemy zaszyfrować jest: PEWNEGO PRZYJACIELA POZNAJE SIE W NIEPEWNEJ SYTUACJI. Usuwamy ze zdania kluczowego spacje i powtarzające się litery i mamy: KRYPTOGAFINLZ. To co otrzymaliśmy ustawiamy na początku alfabetu i w efekcie mamy:

alfabet jawny: **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**

alfabet szyfrowy: **K R Y P T O G A F I N L Z B C D E H J M Q S U V W X**

Na koniec dopasowujemy odpowiednie litery i tworzymy szyfr ( nie zastosowano żadnego przesunięcia). Powinno się uzyskać: DTUBTGC DHXWIKYFTLK DCXBKIT JFT U BFTDTUBTI JWMQKYIF. Deszyfracja odbywa się podobnie.

***Ćwiczenie 6*** Kryptogram UCRTYOKNMCUKHGQZTBMWZQJXKQJMKDFY został zaszyfrowany metodą monoalfabetyczną z wykorzystaniem wyrażenia kluczowego KRYPTOGRAFIA I KRYPTOANALIZA. Zapisz alfabet szyfrowy oraz wiadomość, którą zaszyfrowano.

***Ćwiczenie 7*** Stosując metodę monoalfabetyczną, utwórz alfabet szyfrowy, w którym zdaniem kluczowym będzie powiedzenie: NIE MA REGULY BEZ WYJATKU. Korzystając z utworzonego alfabetu, zaszyfruj tekst: NIE MA RZECZY ZE WSZYSTKICH STRON SZCZESLIWEJ. Zapisz alfabet szyfrowy oraz kryptogram.

**Szyfry wieloalfabetyczne**

Pomimo wprowadzanych utrudnień szyfry monoalfabetyczne nadal nie są trudne do złamania. Dlatego następnym krokiem jest wprowadzenie kilku alfabetów szyfrowych. Jednym z najbardziej znanych szyfrów wieloalfabetycznych jest **szyfr Vigenère’a**. W tej metodzie również stosujemy słowo lub zdanie kluczowe. Najpierw usuwa się z niego spacje i powtarzające się litery. Następnie każda litera klucza staje się początkiem innego alfabetu szyfrowego. Alfabety te tworzy się dopisując kolejne litery alfabetu występujące po tej z klucza.

***Przykład*** Przyjmijmy, że słowem kluczowym jest: TELEFON, a tekstem do zakodowania: NIEOBECNY SAM SOBIE SZKODZI. Po usunięciu powtarzających się liter mamy: TELFON. Każda z liter staje się początkiem innego alfabetu szyfrowego.

Następnym krokiem jest utworzenie szyfru wg zasady: pierwszą literę tekstu jawnego szyfrujemy literą z pierwszego alfabetu, drugą – z drugiego, itd. Po wyczerpaniu alfabetów zaczynamy od nowa aż do zaszyfrowania całego tekstu. W tym przykładzie dostaniemy: GMPTPRVRJ XOZ LSMNS FSOZINV. Deszyfracja odbywa się podobnie.

***Ćwiczenie 8*** Kryptogram QEGTP XPYB PBRECBSYOMM ZFKHSQKKVIK BPORKOE został zaszyfrowany metodą Vigenère’a z wykorzystaniem słowa kluczowego KRYPTOLOGIA. Podaj alfabety szyfrowe i rozszyfrowana wiadomość.

***Ćwiczenie 9*** Stosując szyfr Vigenère’a, utwórz alfabety szyfrowe, w których zdaniem kluczowym jest: NIKT NIE JEST BEZ WAD. Następnie zaszyfruj tekst: GENIUSZ TO TYLKO SPRAWA CIERPLIWOSCI.

***Szyfrowanie asymetryczne***

**Algorytm RSA** ***http://eduinf.waw.pl/inf/alg/001\_search/0067.php***

Jeden z pierwszych i obecnie najpopularniejszych [asymetrycznych](https://pl.wikipedia.org/wiki/Kryptografia_klucza_publicznego) [algorytmów kryptograficznych](https://pl.wikipedia.org/wiki/Szyfr), zaprojektowany w 1977 przez [Rona **R**ivesta](https://pl.wikipedia.org/wiki/Ron_Rivest), [Adi **S**zamira](https://pl.wikipedia.org/wiki/Adi_Szamir) oraz [Leonarda **A**dlemana](https://pl.wikipedia.org/wiki/Leonard_Adleman). Pierwszy algorytm, który może być stosowany zarówno do szyfrowania jak i do [podpisów cyfrowych](https://pl.wikipedia.org/wiki/Podpis_cyfrowy).

Jest to asymetryczny algorytm szyfrujący, którego zasadniczą cechą są dwa klucze: **publiczny** do kodowania informacji oraz **prywatny** do jej odczytywania. Klucz publiczny (można go udostępniać wszystkim zainteresowanym) umożliwia jedynie zaszyfrowanie danych i w żaden sposób nie ułatwia ich odczytania, nie musi więc być chroniony. Dzięki temu firmy dokonujące transakcji poprzez sieć Internet mogą zapewnić swoim klientom poufność i bezpieczeństwo. Drugi klucz (prywatny, przechowywany pod nadzorem) służy do odczytywania informacji zakodowanych przy pomocy pierwszego klucza. Klucz ten nie jest udostępniany publicznie. System RSA umożliwia bezpieczne przesyłanie danych w środowisku, w którym może dochodzić do różnych nadużyć. Bezpieczeństwo oparte jest na trudności znajdowania podzielników dużych liczb - najszybszym komputerom może zajmować to obecnie wiele dziesiątków lat intensywnych obliczeń. Sytuacja prawdopodobnie ulegnie zmianie po zakończeniu sukcesem prac nad komputerami kwantowymi, które będą pracowały miliony razy szybciej od współczesnych komputerów. Ale to dopiero przyszłość...

Algorytm RSA składa się z trzech podstawowych kroków:

1. Generacja klucza publicznego i prywatnego. Klucz publiczny jest przekazywany wszystkim zainteresowanym i umożliwia zaszyfrowanie danych. Klucz prywatny umożliwia rozszyfrowanie danych zakodowanych kluczem publicznym. Jest trzymany w ścisłej tajemnicy.
2. Użytkownik po otrzymaniu klucza publicznego, np. poprzez sieć Internet, koduje przy jego pomocy swoje dane i przesyła je w postaci szyfru RSA do adresata dysponującego kluczem tajnym, np. bank, firma komercyjna, tajne służby rządowe...
3. Adresat po otrzymaniu zaszyfrowanej wiadomości odczytuje ją za pomocą klucza prywatnego.