Kryptografia i kryptoanaliza

Laboratorium 1

Michał Łaskawski

Zadanie 1

Korzystając z języka `C++`, dokonaj implementacji programu szyfrującego i deszyfrującego zadany tekst.

1. Tekst jawny powinien być importowany do programu z pliku tekstowego, którego nazwa określona powinna być po zdefiniowanym argumencie / fladze: `-i`.
2. Wynik pracy programu powinien być eksportowany do pliku tekstowego, którego nazwa określona powinna być po zdefiniowanym argumencie / fladze: `-o`.
3. Klucz powinien być importowany z pliku tekstowego, którego nazwa powinna być określona po zdefiniowanym argumencie / fladze: `-k`.
4. Tryb pracy programu powinien być określony poprzez flagi: `-e` dla procesu szyfrowania, `-d` dla procesu deszyfrowania.

Przykład wywołania programu w celu zaszyfrowania tekstu:

```
./program -e -k klucz.txt -i tekst_jawny.txt -o szyfrogram.txt
```

Przykład wywołania programu w celu odszyfrowania tekstu:

```
./program -d -k klucz.txt -i szyfrogram.txt -o tekst_odszyfrowany.txt
```

Uwagi:

- Kolejność argumentów powinna być dowolna.
- Plik z kluczem powinien mieć formę pliku tekstowego zawierającego definicję tablicy podstawieniowej w postaci dwóch kolumn liter, np:

  ```
  A G
  B O
  C R
  D L
  ...
  Z B
  ```

  Powyższy przykład pokazuje, iż literze `A` przypisana jest litera `G`, literze `B` przypisana jest litera `O` itp.
- Tablica podstawieniowa powinna definiować podstawienia dla wszystkich liter alfabetu łacińskiego (język angielski).
- Tekst jawny (z przeznaczeniem do zaszyfrowania) powinien zawierać dłuższy akapit a lepiej kilka akapitów pisanych w języku angielskim (najlepiej beletrystyka).
- Odczytany tekst jawny, przed dalszym przetwarzaniem, powinien być zamieniony do postaci składającej się tylko z dużych liter. Ponadto z tekstu powinny być usunięte wszystkie znaki, które nie są literami, np: odstępy, przecinki, kropki itp.

Zadanie 2

Rozbudować program z poprzedniego przykładu poprzez dodanie do niego funkcjonalności generowania statystyk liczności występowania `n-gramów` (sekwencji kolejnych liter), to jest `mono-gramów` (pojedynczych liter), `bi-gramów` (wyrazów dwuliterowych), `tri-gramów` (wyrazów trzyliterowych) oraz `quad-gramów` (wyrazów czteroliterowych). Funkcjonalność ta powinna być wyzwalana poprzez dodanie do programu jednej z następujących flag: `-g1`, `-g2`, `-g3` lub `-g4`, po której powinna zostać określona nazwa pliku, do którego zapisane zostaną wyniki.

Przykład wywołania programu:

```
./program -i tekst_jawny.txt -g1 monogramy.txt
```

Przykład wyznaczania `bi-gramów` dla tekstu:

Tekst jawny:

```
This is an example of plain text
```

Tekst wstępnie przetworzony:

`THISISANEXAMPLEOFPLAINTEXT`

Kilka pierwszych `bi-gramów`:

```
1. TH
2. HI
3. IS
4. SI
5. IS
6. SA
...
```

Dla każdego wyznaczonego `n-gramu` należy wyznaczyć liczność jego występowania w badanym tekście. Wynik pracy programu powinien być wygenerowany w postaci tabeli:

`n-gram liczbość`

Przykład:

```
TH 1
HI 1
IS 2
SI 1
SA 1
...
```

Zadanie 3

Uzupełnij program z poprzedniego zadania, tak aby w przypadku podania flagi `-rX`, gdzie `X` jest liczbą należącą do zbioru `{1, 2, 3, 4}` a następnie nazwy pliku, program odczytywał z niego referencyjną bazę `n-gramów`. Liczby z podanego zbioru odpowiadają: `{mono-gramom, bi-gramom, tri-gramom, quad-gramom}`.

Uwaga: Odczytana referencyjna baza `n-gramów` powinna być tabelą, której każdy wiersz składa się z dwóch wartości oddzielonych spacją. Wartościami tymi powinny być: G_i oraz P_i, gdzie G_i to i-ty `n-gram` natomiast P_i to prawdopodobieństwo wystąpienia i-tego `n-gramu` w tekście referencyjnym (*corpus*). Prawdopodobieństwo to wyznaczyć można korzystając ze wzoru: $P_i = N_i/N$, gdzie N_i jest licznością wystąpienia danego `n-gramu` w tekście referencyjnym, natomiast N jest całkowitą liczbą wszystkich `n-gramów` w tekście referencyjnym.

Następnie należy rozbudować program, tak aby podanie flagi `-s` generowało wartość testu χ^2 dla zadanego tekstu (flaga `-i`) i wybranej bazy referencyjnej (flaga `-rX`). Wynik działania programu powinien być drukowany na standardowe wyjście.

W kontekście zadania, test χ^2 może być zdefiniowany następująco: $T = \sum_{i=0}^{n} \frac{(C_i - E_i)^2}{E_i}$, gdzie: C_i jest liczbą wystąpień i-tego symbolu (`n-gramu`) w analizowanym tekście, $E_i = n \cdot P_i$ jest oczekiwaną liczbą wystąpień i-tego symbolu (`n-gramu`) w tekście, n jest całkowitą liczbą `n-gramów` w analizowanym tekście.

Zadanie 4

Wykonać eksperymenty:

- Dokonaj obserwacji wyniku testu χ^2 dla tekstu jawnego i zaszyfrowanego o różnych długościach.
- Wiadomo, iż wynik testu może być znacząco zaburzony w przypadku gdy brane są pod uwagę symbole (`n-gramy`), które rzadko występują w tekście, np w przypadku `mono-gramów` języka angielskiego są to litery: `J`, `K`, `Q`, `X` oraz `Z` (patrz odczytana tablica częstości `mono-gramów`). Zbadaj wynik testu χ^2 w przypadku gdy do wyznaczenia testu pominięte zostaną rzadko występujące `n-gramy`.