KRIPTOGRAFIJA
Zadaća 2.130

Filip Nikšić

9. travnja 2007.

1. Vigenèreovom šifrom iz otvorenog teksta na hrvatskom jeziku dobiven je šifrat:

```
WLRMG JKCOK FLLET SAACV EEDUM JRRIC FIEZH SMRWB
KEFHT WOUHH KOKGB DIJPH HSZHT PBAUO MJVHX VJLNT
SNAYF MIJNN BLZOL LOBDX QOEIO PPFNO SDZID BKFCL
URRTN KEEUO PDEOH UMZWN
```

Odredite najprije duljinu ključne riječi, potom samu ključnu riječ, te dekriptirajte šifrat.

Rj. Zadatak sam riješio programom *vigenere.pl* napisanim u Perlu. Rješavanje se sastoji od nekoliko faza.

Prvo se Friedmanovim testom određuje duljina ključa. Evo koda:

```
sub duljina_kljuca {
    our @sifrat;
    our %analiza;

    for (my $m=3; $m<=15; $m++) {
        $analiza{$m}=[];
        print "m=$m: ";
        for (my $pocetak=0; $pocetak<$m; $pocetak++) {
            my %hash=();
            my ($ic, $n)=(0.0, 0);
            for (my $i=$pocetak; $sifrat[$i]; $i+=$m, $n++) {
                $hash{$sifrat[$i]}++;
            }
            push @{$analiza{$m}}, \%hash;
            foreach (keys %hash) {
                $ic+=$hash{$_}*($hash{$_}-1);
            }
            $ic/=$n*($n-1);
            printf " %.3f", $ic;
        }
        print "\n";
    }
}
```

Nakon što za određenu duljinu ključa `$m` seciram `@sifrat` na `$m` podnizova, analizu slova podnizova pohranjujem u `$m` hash tablica oblika `slovo` $\mapsto$ `frekvencija`. Polje s referencama na te hash tablice pridruzujem ključu `$m` u globalnoj hash tablici `%analiza`.

Program za svaki `$m` izračuna odgovarajuće indekse koincidencije i ispiše ih. Za moj šifrat program ispisuje sljedeće:

```
m=3:  0.042 0.032 0.036
m=4:  0.037 0.047 0.034 0.035
m=5:  0.048 0.053 0.069 0.053 0.063
```

```
m=6:  0.047 0.033 0.043 0.028 0.028 0.032
m=7:  0.037 0.016 0.053 0.037 0.026 0.021 0.058
...
```

Ostale duljine nisu previše bitne, jer za `$m` == 5 uočavam pristojne indekse koincidencije (ako ništa, pristojnije nego u ostalim slučajevima).

U sljedećoj fazi program od korisnika traži da se odluči za jednu duljinu ključa s kojom se nastavlja. Kod mene ta duljina je 5. Evo i koda:

```
sub obradi_duljinu {
    our %analiza;
    my %mics;

    print "\nm=";
    my $m=<STDIN>;
    chomp $m;

    foreach my $aref (@{$analiza{$m}}) {
        for (my $pomak=0; $pomak<26; $pomak++) {
            my ($mic, $n)=(0.0, 0);
            foreach my $key (keys %$aref) {
                my $slovo=chr((ord($key)-ord('a')-$pomak)%26+ord('a'));
                $mic+=$freq_hrv{$slovo}*$aref->{$key};
                $n+=$aref->{$key};
            }
            $mic/=$n;
            $mics{chr($pomak+ord('a'))}=$mic;
        }
        foreach my $key (sort {$mics{$b}<=>$mics{$a}} keys %mics) {
            last if ($mics{$key}<0.05);
            printf " %s-%.3f", $key, $mics{$key};
        }
        print "\n";
    }
}
```

Nakon korisnikovog unošenja duljine ključa, program ponovno koristi analize slova napravljene u prvoj proceduri. Vanjska petlja prolazi svim referencama hash tablica spremljenih u polju `@{$analiza{$m}}`. Te hash tablice sadrže podatke o frekvencijama slova u svakom od `$m` podnizova.

Za svaku referencu `$aref`, unutarnja petlja računa međusobne indekse koincidencije za svaki `$pomak`. Pri računanju, koriste se relativne frekvencije slova u hrvatskom jeziku spremljene u globalnu hash tablicu `%freq_hrv`. Izračunati indeks pridruži se `$pomak`-u pretvorenom u slovo u hash tablici `%mics`.

Konačno, procedura za svaki podniz ispiše pomake za koje je međusobni indeks koincidencije veći ili jednak 0.05, tako da lako možemo uočiti ključ. Evo ispisa za moj slučaj:

```
 b-0.059 s-0.052
 a-0.061
 r-0.072 w-0.051
 u-0.062 z-0.051
 t-0.067
```

Očito je da je ključ `barut`.

Posljednja faza rješavanja je dešifriranje šifrata. To radi sljedeća procedura:

```
sub desifriraj {
    our @sifrat;

    print "\nUnesi kljuc: ";
    my $input=<STDIN>;
    chomp $input;
    my @kljuc=split '', $input;

    foreach (@sifrat) {
        my $slovo=chr((ord($_)-ord($kljuc[0]))%26+ord('a'));
        print $slovo;
        push @kljuc, shift @kljuc;
    }
    print "\n";
}
```

Od korisnika se traži unos ključa koji prepozna u prethodnoj fazi. Kod više-manje objašnjava sam sebe. Naredba `push @kljuc, shift @kljuc;` rotira ključ tako da obriše prvo slovo (`shift`) i stavi ga na posljednje mjesto (`push`).

Evo ispisa za moj slučaj, uz ključnu riječ `barut`:

```
vlasniklurelukarajicdematiraojeinformacijeonavodnojotmicisvogsinaobjav
ljeneujutarnjemlistualiuskokjeponovopotvrdiokakoistrazujenavodnuotmicu
```

Ljepše napisan, otvoreni tekst glasi:

Vlasnik Lure, Luka Rajić, dematirao je informacije o navodnoj otmici svog sina objavljene u Jutarnjem listu, ali USKOK je ponovo potvrdio kako istražuje navodnu otmicu.

2. Dešifrirajte šifrat:

```
REOFX RLRHU CHFSB GDUEP RVBFB GDUEP RVYVS EAOGD
KRATR VRMJT RVASY JPFCN NUPGD GRXAS FNBMN USARX
OFASO BPYNU CUKHD GRXOF CASOL RZOTK PFRSO BHAFS
DGYJA SBSRX SORL
```

šifriran Playfairovom šifrom s ključnom riječi "HONDURAS".

Rj. Pomoću ključne riječi "HONDURAS" konstruiramo matricu slova:

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

Dešifriranje provodimo u blokovima od po dva slova, na sličan način na koji se provodi šifriranje Playfairovom šifrom. Jedina razlika je što se u slučaju dva slova u istom retku, odnosno stupcu, pomičemo ciklički za jedno mjesto ulijevo, odnosno prema gore.

Otvoreni tekst glasi:

```
hrvat skaud rugas indik atais indik atxtr govin
ehrva tskez atraz ilisu odmin istra gospo darst
varad aipod uzetn istva brank avuke licad aorga
nizir asast anak
```

Ljepše napisano:

Hrvatska udruga sindikata i Sindikat trgovine Hrvatske zatražili su od ministra gospodarstva, rada i poduzetništva, Branka Vukelića, da organizira sastanak.

3. Odredite ključ K u Hillovoj šifri ako je poznato da je $m = 2$, te da otvorenom tekstu

```
uskok nasto jipro vjeri tiinf o-- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --
```

odgovara šifrat

```
SGAGU VEQFI BMPRY DDURI VUQJD CPAMW SROWN AJUTF
EBMHU EAWZK QFKYQ JDCPA MWSRU CYFEQ XKJOJ MESWU
LSIWG BIGIG CTCAM UKAMX ALOBK BYP
```

Dešifrirajte ostatak poruke.

Rj. U Hillovoj šifri imamo sljedeću situaciju:

$$y = xK$$

gdje x predstavlja blok otvorenog teksta, K matricu ključa, a y odgovarajući blok šifrata. Kod nas su blokovi duljine $m = 2$, dakle, vektori su dimenzije 2, a K je 2×2 matrica.

Kombinirajući dva bloka otvorenog teksta i odgovarajućih blokova šifrata, dobivamo matričnu jednadžbu:

$$Y = XK$$

Ukoliko je matrica X invertibilna nad prstenom $\mathbb{Z}_{26}$, lako možemo izračunati K:

$$K = X^{-1}Y$$

Pokušajmo pronaći dva bloka otvorenog teksta za koje je matrica X invertibilna:

$$\begin{aligned} X &= \begin{bmatrix} \texttt{u} & \texttt{s} \\ \texttt{k} & \texttt{o} \end{bmatrix} = \begin{bmatrix} 20 & 18 \\ 10 & 14 \end{bmatrix} \implies \det X = 2 \\ X &= \begin{bmatrix} \texttt{k} & \texttt{n} \\ \texttt{a} & \texttt{s} \end{bmatrix} = \begin{bmatrix} 10 & 13 \\ 0 & 18 \end{bmatrix} \implies \det X = 24 \\ X &= \begin{bmatrix} \texttt{t} & \texttt{o} \\ \texttt{j} & \texttt{i} \end{bmatrix} = \begin{bmatrix} 19 & 14 \\ 9 & 8 \end{bmatrix} \implies \det X = 18 \end{aligned}$$

Nažalost, sve dobivene determinante su parne, a parni brojevi nemaju inverz u $\mathbb{Z}_{26}$. Mi želimo dobiti neparnu determinantu. Odaberimo onda među već isprobanim blokovima one za koje će umnožak elemenata dijagonale biti neparan, a umnožak ostalih elemenata paran. Npr.

$$X = \begin{bmatrix} \texttt{t} & \texttt{o} \\ \texttt{k} & \texttt{n} \end{bmatrix} = \begin{bmatrix} 19 & 14 \\ 10 & 13 \end{bmatrix} \implies \det X = 3$$

Sad imamo $(\det X)^{-1} = 9$ pa je

$$X^{-1} = (\det X)^{-1} \begin{bmatrix} 13 & -14 \\ -10 & 19 \end{bmatrix} = \begin{bmatrix} 13 & 4 \\ 14 & 15 \end{bmatrix}$$

Matricu Y dobijemo tako da pogledamo u što se šifriraju blokovi `to` i `kn`:

$$Y = \begin{bmatrix} \texttt{F} & \texttt{I} \\ \texttt{U} & \texttt{V} \end{bmatrix} = \begin{bmatrix} 5 & 8 \\ 20 & 21 \end{bmatrix}$$

Konačno, izračunamo K:

$$K = X^{-1}Y = \begin{bmatrix} 15 & 6 \\ 6 & 11 \end{bmatrix}$$

Lako se provjeri da je matrica K involutorna pa je sama sebi inverz. Prema tome, imamo:

$$x = yK$$

Sad idemo redom, blok po blok šifrata i provodimo dešifriranje. Rezultat:

```
uskoknastojiprovjeritiinformacijeonavodnojotmicitakvim
informacijamaraspolazemoionesesvakakonemoguignoriratix
```

Ljepše napisano:

USKOK nastoji provjeriti informacije o navodnoj otmici. Takvim informacijama raspolažemo i one se svakako ne mogu ignorirati.