1. Domača naloga

Matija Kerkoč

March 25, 2019

Opisi funkcij so podani v python datoteki, kjer so funkcije definirane, v dokumentu bomo opisali ideje, ki se skrivajo za njimi.

1. Naloga

Funkciji $Encrypt(b, k)$ ter $Decrypt(c, k)$ delujeta kot v navodilih. Ideja je, da s pomočjo slovarja črk angleške abecede, ki jim priredimo zaporedno številko $"A" : 0, \ldots, "Z" : 25$ preračunamo ter nato kriptiramo oz. odkriptiramo besedilo.

Ideja rešitve je sledeča:

1. poiskati dolžino ključa,
2. določiti kateri ključ je najboljši,
3. odkriptirati besedilo.

1. Glavna funkcija tukaj je $dolzina\_kljuca(c)$. Ideja je, da v besedilu c pregledamo vse možne nize zaporednih črk dolžine 3 (oz. lahko tudi 4 ali več, če se ponovita le dve črki potem je to bolj slučaj, kot smo povedali na predavanjih) ter v slovar zapišemo razmake med njimi. Nato v slovar zapišemo vse delitelje razmakov med vsemi možnimi nizi oz povečamo števe pri deliteljih, ki so že v slovarju. Na koncu iz slovarja dobimo nekaj najprimernejših kandidatov.

2. Glavni problem je, če imamo ključ dolžine npr. 2, ali je mogoče pravilni ključ dolžine 4? Če se število deliteljev teh dveh ključev ne razlikuje "preveč" ali je pravilen daljši ključ? V splošnem je lahko pravilen ključ, katerega dolžina ni nujno najpogostejša med delitelji, vendar je še vedno dovolj pogost. Ta problem rešimo tako, da rešitev izračunamo za nekaj najpogostejših ključev, nato pa v končnem odkriptiranem besedilu iščemo nekaj najpogostejših besed angleške abecede ter vrnemo najprimernejše besedilo.

3. Glavna funkcija je $ujemanje(c)$. Funkcija vzame vsako n-to črko, kjer je n dolžina ključa, ter vrne odkriptirano vsako n-to črko. Tukaj se sedaj pojavi problem s primerjanjem. Problem rešimo tako, da za vsako črko abecede konstruiramo pomik ter zanjo zapišemo slovar deležev črk. Sedaj za vsako izmed črk po katerih smo iterirali izračunamo ujemanje abeced. To stori funkcija

$najmanjsi(s1, s2)$, kjer smo vzeli kvadrate razlik med deleži posameznih črk. Zapišemo seznam vseh indeksov ujemanj ter vrnemo odkriptirano besedilo, ki ga porodi najboljši približek za zamik.

Sedaj to ponovimo, za vsako izmed mest v ključu.

Besedilo odkriptiramo s pomočjo funkcije $razbij\_Viginer(c)$, ki nam vrne odkriptirano besedilo.

2. Naloga

Rešitev zopet sestoji iz treh najpomembnejših poddelov:

1. iskanje inverza matrike v $\mathbb{Z}_{26}$,
2. izračun inverza v $\mathbb{Z}_{26}$,
3. odkriptirati besedilo.

1. Definiramo pomožni funkciji $razsirjeni\_evklid(a, b)$, ki nam izračuna razširjeni Evklidov algoritem ter $inverz\_po\_modulu(a, m)$, ki nam izračuna inverz števila $a \pmod m$. Definiramo tudi množenje matrike z vektorjem ter množenje matrik po modulu 26. Vse te funkcije so le pomožne funkcije za $inverz\_matrike(A)$, ki izračuna inverz matrike v $\mathbb{Z}_{26}$.

2. Funkciji $razbij(c)$ ter $ugani\_kljuc(c)$ iz kriptograma c dobita najprej nekaj najpogostejših dvojic besed v angleških besed, ter jih nato za vsako možno kombinacijo dvojic tudi primerjata. $ugani\_kljuc(c)$ nato vrne seznam vseh možnih besedil.

3. Ostane nam le še to, da s funkcijo $Decrypt(c, k)$ odkriptiramo besedilo za vse potencialne ključem nato pa $razbij\_Hill(c)$ s pomočjo slovarja najpogostejših besed angleške abecede poišče odkriptirano besedilo, ki je res pravilna rešitev.

Odkriptirano besedilo iz naloge:

ITISAVERYPOORTHINGWHETHERFORNATIONSORINDIVIDUALS
TOADVANCETHEHISTORYOFGREATDEEDSDONEINTHEPASTASAN
EXCUSEFORDOINGPOORLYINTHEPRESENTBUTITISANEXCELLENT
THINGTOSTUDYTHEHISTORYOFTHEGREATDEEDSOFTHEPAST
ANDOFTHEGREATMENWHODIDTHEMWITHANEARNEST
DESIRETOPROFITTHEREBYSOASTORENDERBETTERSERVICEINTHE
PRESENTINTHEIRESSENTIALSTHEMENOFTHEPRESENT
DAYAREMUCHLIKETHEMENOFTHEPASTANDTHELIVE
ISSUESOFTHEPRESENTCANBEFACEDTOBETTERADVANTAGE
BYMENWHOHAVEINGOODFAITHSTUDIEDHOWTHELEADERSOF
THENATIONFACEDTHEDEADISSUESOFTHEPASTSUCHASTUDYOF
LINCOLNSLIFEWILLENABLEUSTOAVOIDTHETWINGULFSOF
IMMORALITYANDINEFFICIENCYTHEGULFSWHICHALWAYS
LIEONEONEACHSIDEOFTHECAREERSALIKEOFMANANDOFNATION
ITHELPSNOTHINGTOHAVEAVOIDEDONEIFSHIPWRECK
ISENCOUNTEREDINTHEOTHER

Zaključek

Pri obeh nalogah je v rešitev dodana funkcija, ki za več možnih odkriptiranih besedil poišče pravo. To stori s pomočjo slovarja najpogostejših angleških besed, ter za vsakega izmed možnih kandidatov poišče tistega, ki se najbolje ujema. Tako na preprost način računalniku povemo, kaj je *smiselno* besedilo in ne le nekaj naključnih črk.

Program v taki obliki ne bi deloval tudi za matrike višjih dimenzij, vendar pa bi bila ideja rešitve enaka. Definirati bi bilo potrebno funkcije, ki bi izračunale inverz matrike $n \times n$, množenje matrik $n \times n$, ter nato iskati ujemanja z angleško abecedo v nizih dolžine $n \times n$. Osnovna ideja ostaja ista: poiskati najpogostejše nize dolžine $n \times n$ ter ugotoviti, kateri se ponavljajo najpogosteje, nato pa le te primerjati z nizi dolžine $n \times n$ v angleški abecedi.

Če bi nam uspelo najti nek dovolj enostaven način za definicijo množenja, inverza ter množenja matrik $n \times n$ z vektorji, potem bi funkcije, ki res odkriptirajo besedilo le nekoliko popravili in bi naša rešitev delovala tudi za matrike večjih dimenzij. Vendar pa so omenjene funkcije že precej težke za definicijo (predvsem je tukaj najtežje napisati funkcijo za inverz matrike).