

# Princip rada koder i dekodera

Uros Milovanovic

June 6, 2021

## 1 Bijekcija

Za kodiranje podataka znacajan je pojam bijektivne funkcije. Cilj nam je da preslikamo skup karaktera iz ASCII tabele u samog sebe. To je konacan skup, pa nam bijektivno preslikavanje govori o tome da je svaki element iz skupa karaktera preslikan jednoznacno, na tacno jedan element istog skupa. Tako i svaka slika ima tacno jedan original.

Ovakve funkcije se koriste za kodere i dekodere. Funkcija koja definise dekodier jeste inverzna funkcija od one koja definise koder. Buduci da isključivo bijektivne funkcije imaju svoje inverzne funkcije, neophodno je da koder i dekodier budu tako definisani.

## 2 Koder

Prosirena ASCII tabela jeste tabela u kojoj je svakom karakteru, kao i određenom broju naredbi dodeljen broj. Ukupan broj karaktera i naredbi je 256. Naredbe su numerisane brojevima 0 – 31, dok su karakteri numerisani brojevima 32 – 255. Stoga pravimo bijekciju iz skupa  $\{x \mid x \in N, 0 \leq x \leq 255\}$  u samog sebe.

Primetimo da se svaki član ovog skupa jednoznacno može predstaviti u obliku

$$16 \cdot a + b, \quad a, b \in N, \quad 0 \leq a, b \leq 15$$

Dakle, svaki član skupa karaktera možemo predstaviti uredjenim parom  $(a, b)$ . Npr.  $0 = (0, 0)$ ,  $1 = (0, 1)$ ,  $16 = (1, 0)$ ,  $19 = (1, 3)$ ... Sustina kodiranja jednog karaktera je sledeca:

- Uzimanje odgovarajuceg uredjenog para od clana skupa karaktera
- Zamena mesta clanovima uredjenog para
- Prevod novog uredjenog para u broj
- Dodavanje konstante na novi broj
- Uzimanje ostatka novog broja pri deljenju sa 256.

Primer:

- $46 = 2 \cdot 16 + 14$ , dakle,  $46 = (2, 14)$
- $(2, 14)$  prelazi u  $(14, 2)$
- $(14, 2) = 14 \cdot 16 + 2 = 226$
- neka je konstanta  $c = 12$ . Imamo  $226 + 3 = 229$
- $229 \bmod 256 = 229$

Poslednji korak se izvrsava zato sto dodavanjem konstante mozemo premasiti broj 255 i time izaci iz skupa karaktera.

Kod svakog kodiranja u zadatom skupu postoji problem. Neki karakteri se kodiraju u komande, sto onemogucava pravilan ispis koda. Ovaj problem se moze resiti na mnogo nacina. U primenjenoj varijanti karakter koji bi se kodirao u komandu ostaje nepromenjen, a u drugim, umetnutim karakterima, cuvamo informaciju o tome da li je program naisao na takav karakter. Sustina kodiranja citave poruke je sledeca:

- Kodiranje jednog karaktera
- Postavljanje dva nasumicna karaktera u odredjenom opsegu nakon svakog kodiranog karaktera
- Ukoliko se karakter preslikava u komandu, nakon njega karakter nije nasumican, vec uzima specificnu vrednost van gore pomenutog opsega

### 3 Dekoder

Funkcija koja odredjuje koder je inverzna funkcija koderskoj. Od kodiranog karaktera, tj. kodirane poruke, potrebno je izvuci poslatu informaciju.

Zbog strukture koda, znamo da je kodirana poruka tri puta duza od prave, i da je, pocevsi od prvog, svaki treci karakter deo prvobitne poruke. Dekodiranje karaktera se sastoji iz sledeceg:

- Posmatranje svakig treceg karaktera, pocevsi od prvog
- Ukoliko karakter nakon njega ima specificnu vrednost, karakter koji se posmatra je isti onakav kakav je bio u prvobitnoj poruci
- Ako to nije slucaj, najpre se oduzima konstanta
- Uzima se ostatak tog broja pri deljenju sa 256, i ako je taj ostatak negativan, dodaje se broj 256 na njega
- Uzima se uredjeni par koji odgovara dobijenom broju i vrsi ista procedura sa zamenom mesta clanova para kao pri kodiranju