# Zaštitno kodiranje signala

## Laboratorijska vježba 4

SP

*Metode optimalnoga i ravnomjernoga kôdiranja (Shannon-Fano kôd, Huffmanov kôd, Hammingov kôd)*

Student:

| | | |
|---|---|---|
| prezime | Ime | mat. broj |
| | Laboratorij | Datum |

# Sadržaj:

Sazdano od:

Definicije:

***Optimalno kodiranje*** … postupak kodiranja koji ostvaruje minimalno vrijeme prijenosa poruka.

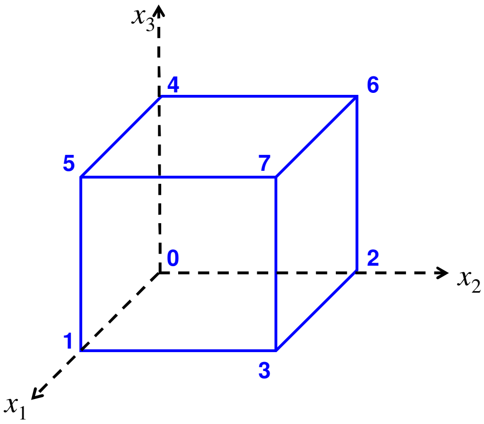
***Ravnomjerno kodiranje*** … postupak kodiranja kod kojega su sve kodne riječi jednake dužine.

Telekomunikacije\Teorija informacija 1\Optimalan kôd.doc (cjelovit primjer u 18 slika nalazi se na MOODLE)

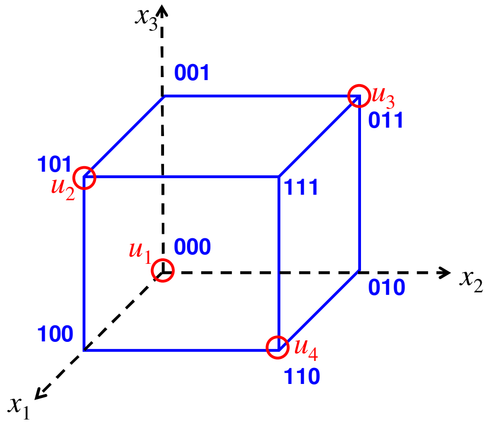
# 4. 4. METODE RAVNOMJERNOGA I OPTIMALNOGA KÔDIRANJA (SHANNON-FANO KÔD, HUFFMANOV KÔD, HAMMINGOV KÔD)

## 4.1. 4.1. Ravnomjerno kodiranje

Zbog lakše predodžbe primijenit ćemo geometrijsku predstavku kodova. Kodna riječ koja se sastoji od $K_u$ simbola (bitova), može se prikazati koordinatama $d_v$ (koje mogu imati vrijednosti 0 ili 1) u $K_u$ dimenzijskom prostoru $E_{K_u}$. Ako je $K_u = 3$, prostor se može sasvim pregledno prikazati u tri dimenzije. Ako s $x_1$, $x_2$ i $x_3$ označimo koordinate prostora $E_3$, tada se vrhovi kocke mogu označiti brojevima 0, 1, 2, …, 7. Ako svakome od tih brojeva pridijelimo normalan binaran ekvivalent, tada će npr. koordinate vrha 7 biti: $x_1 = 1$, $x_2 = 1$ i $x_3 = 1$ (slika 4.1).




*Slika 4.1 Geometrijski prikaz kôda*




*Slika 4.2 Sigurnosni kôd s udaljenošću 2.*

Želimo li pomoću kôda s $K_u = 3$ (bita) kodirati četiri vijesti, $N = 4 = \{u_1, u_2, u_3 \text{ i } u_4\}$, tada će u kôdu postojati zalihost:

$$r = \frac{K_u - \mathrm{ld}\,N}{\mathrm{ld}\,N} = \frac{K_u - K}{K} \Rightarrow r = \frac{3-2}{2} = 0{,}5 = 50\%$$

Naš cilj je da tu zalihost od 50% iskoristimo za sigurniji prijenos informacija.

Gledano geometrijski, potrebno je na pogodan način odabrati one vrhove kocke, u koje ćemo smjestiti naše četiri vijesti. U tu svrhu uvedimo pojam *udaljenosti dvaju vrhova*, kao *najmanji broj bridova* koji ih dijele. Npr. udaljenost vrhova 1 i 7 na slici 4.1 jednaka je 2. Udaljenost između dva vrha označavat ćemo s $d_{ij}$. Ako sada pridijelimo informacijama one vrhove kocke čija udaljenost je 2, moći ćemo načiniti kôd u kojemu je moguće otkriti jednostruku pogrešku. Sasvim je jasno da je moguće načiniti više takvih kodova zavisno od toga koji vrh odaberemo kao početni. Jedna mogućnost prikazana je na slici 4.2.

U tablici dolje, dât je odgovarajući *ravnomjeran* kôd.

| vijest | $u_1$ | $u_2$ | $u_3$ | $u_4$ |
|---|---|---|---|---|
| kôd | 000 | 101 | 011 | 110 |

Svojstvo $d_{ij} = 2$ u ovome slučaju znači, da su kôdne kompleksije tako odabrane, da se one razlikuju u dvije znamenke. Drugim riječima, nastupi li prilikom prijenosa jednostruka pogreška u kôdnoj riječi, time nastaje *nova kôdna riječ*, koja nije pridodana ni jednoj drugoj vijesti (zalihosna riječ).

Npr. ako kod prijenosa vijesti $u_3$ nastupi pogreška, te se kôdna kompleksija primi kao 001, prijemnik će detektirati pogrešku, ali neće moći ustanoviti koja je vijest pravilna, jer je primljena kompleksija 001 mogla nastati ili od 101 ili od 011 ili od 000. Jasno je da će dvostruka pogreška u jednoj riječi pretvoriti je u drugu, koja je također pridodana jednoj od vijesti. Prema tome takav kôd omogućuje *otkrivanje* ali ne i *ispravak* jednostruke pogreške.

Razmotrimo dalje slučaj, kada istim kôdom želimo kodirati samo dvije vijesti. Tada je $r = 100\%$. U tome slučaju možemo odabrati $d_{ij} = 3$. Sada će jedna pogreška u kôdnoj riječi dati riječ koja je od izvorne udaljena za $d$ = l, a od svih drugih, koje su primijenjene za kodiranje (u ovome slučaju je to samo jedna) za $d$ = 2. Prema tome takav kôd imat će mogućnost:

1. ispraviti jednu pogrešku. Npr. odaberimo kôd prema tablici:

| vijest | $u_1$ | $u_2$ |
|---|---|---|
| kôd | 000 | 111 |

Ako je primljena riječ 101, očito je pravilna informacija 111, uz *pretpostavku da je nastupila samo jedna pogreška*, jer kompleksije 001 i 100 u odabranome kôdu nemaju značenje vijesti.

2. otkriti dvije pogreške, bez mogućnosti ispravke.

Na primjer riječ 100 mogla je nastati od riječi 111 uz dvostruku pogrešku ili od riječi 000 uz jednostruku pogrešku. Prema tome možemo samo spoznati nastup pogreške, ali ne možemo ustanoviti pravilnu informaciju.

Svojstva kodova razmatrana na $K_u$ = 3 možemo primijeniti na opći slučaj. Dakako da je tada geometrijski prikaz složeniji i zahtijeva uvođenje višedimenzijskih prostora. Općenito, ako je u $K_u$ dimenzijskome prostoru $E_{K_u}$ udaljenost između dviju točaka, koje predstavljaju niz od $N$ simbola 0, 1, …, $N$–1, poredanih u prirodnome poretku, jednaka $d$, tada kôdne kompleksije što su iskorištene za kodiranje simbola, čine podprostor u kojemu je udaljenost između ma kojih dviju točaka jednaka $d$.

Prema tome vrijedi za bilo koje dvije kôdne riječi $M_i$ i $M_j$, da je $d_{ij} \geq d$. Ako u kôdnoj riječi nastupi jedna pogreška, tada će to dovesti do pogrešne riječi $M_i^{'}$, koja je udaljena od pravilne za jedinicu, a od svih ostalih riječi za najmanje $d - 1$. Iz toga proizlazi da je očito moguće otkriti jednostruku pogrešku čiji nastup dovodi do pojave točke koja je za α udaljena od riječi $M_i$, ako je ispunjen uvjet:

$$d - \alpha \geq 1 .$$

Na taj način vidimo da je maksimalan broj pogrešaka koje možemo otkriti jednak $d$–1. Takav kôd može dakle otkriti prisutnost $d$–1 pogrešaka, ali pri tomu može ispraviti samo $\frac{d}{2}-1$ pogrešku ako je $d$ parno, ili $\frac{d-1}{2}$ ako je $d$ neparno. Za neke vrijednosti udaljenosti, tablica pokazuje koliko je moguće pogrešaka otkriti i ispraviti.

| Minimalna vrijednost udaljenosti $d$ | Mogućnosti | |
|---|---|---|
| | otkrivanja | ispravak |
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 2 | 1 |
| 4 | 3 | 1 |
| 5 | 4 | 2 |

Iz gornjih razmatranja proizlazi, da je uvijek moguće binarno kodiranje informacija takvim kôdom, koji u sebi sadrži svojstva što omogućuju i *otkrivanje* i *ispravak* pogrešaka.

Postoji i niz drugih metoda za sintezu sigurnosnih kodova od kojih je najpoznatija Hammingova. Uvid u tu metodu najlakše se dobije preko primjera.

### 4.1.1. 4.1.1. RAVNOMJERNI KODOVI

Uzmimo npr. da želimo kodirati znakove alfabeta hrvatskoga jezika. Tu je $N$ = 27. Odaberemo li binarni kôd, tada je $L$ = 2. Da bi ispunili uvjet (2.50) mora biti $n$ = 5 tj. $2^5 = 32 > 27$. Prema tome, sve znakove možemo predstaviti različitim kodnim riječima, koje su sastavljene od pet binarnih elementarnih simbola (0 i 1). Jedan od takvih kodova je i internacionalni kôd broj 2, prikazan sljedećom tablicom:

Tablica 2.1.

| ZNAK | KOD 12345 | | ZNAK | KOD 12345 | | ZNAK | KOD 12345 | | ZNAK | KOD 12345 |
|---|---|---|---|---|---|---|---|---|---|---|
| A | 11000 | | Q | 11101 | | I | 01100 | | Y | 10101 |
| B | 10011 | | R | 01010 | | J | 11010 | | Z | 10001 |
| C | 01110 | | S | 10100 | | K | 11110 | | | 00010 |
| D | 10010 | | T | 00001 | | L | 01001 | | | 01000 |
| E | 10000 | | U | 11100 | | M | 00111 | | Slova | 11111 |
| F | 10110 | | V | 01111 | | N | 00110 | | Brojke | 11011 |
| G | 01011 | | W | 11001 | | O | 00011 | | Razmak | 00100 |
| H | 00101 | | X | 10111 | | P | 01101 | | Kombinacija 32 | 00000 |

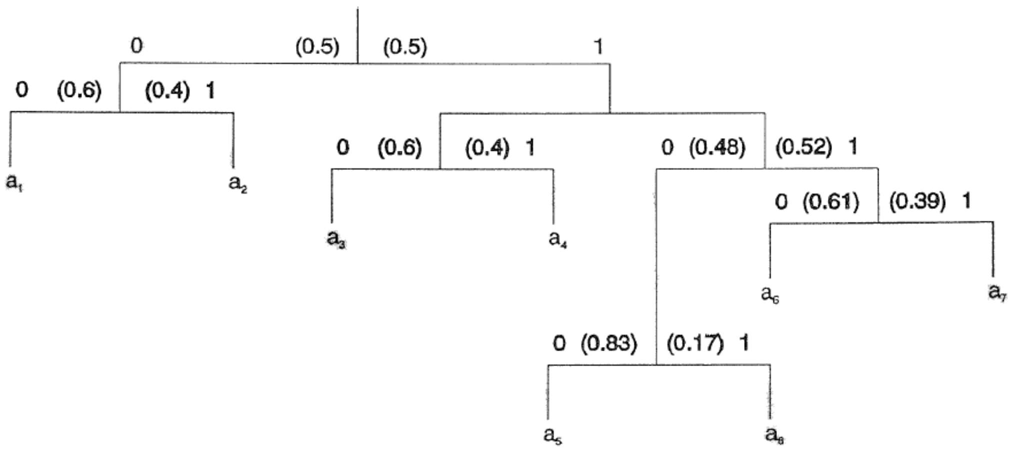
Ako u fizikalnoj reprezentaciji kôda, oba elementarna simbola (strujni impuls i pauza) imaju jednako trajanje, tada da sve kôdne riječi imati jednaku dužinu. Takvi kodovi nazivaju se *ravnomjernima*. Usporedimo li takve kodove s *neravnomjernim*, tada ovi prvi imaju tehničke prednosti, tj. jednostavni su za dekodiranje.

## 4.2. 4.2. Optimalno kodiranje

Cjelovit primjer optimalnoga kodiranja u 18 slika, nalazi se na MOODLE)

### 1. *PRIMJER 4.1: Jedan način optimalnoga kodiranja*

Zadano je 8 poruka s pripadnim vjerojatnostima. Slika 4.3 ilustrira moguće stablo kodiranja koje proizlazi iz kriterija da su vjerojatnosti grana svakoga čvorišta što bliže 0.5.




*Slika 4.3 Kodno stablo.*

| A | $p_i$ | kôd |
|---|---|---|
| $a_1$ | 0.3 | |
| $a_2$ | 0.2 | |
| $a_3$ | 0.15 | |
| $a_4$ | 0.1 | |
| $a_5$ | 0.1 | |
| $a_6$ | 0.08 | |
| $a_7$ | 0.05 | |
| $a_8$ | 0.02 | |

Već na jednostavnom primjeru 4.1, uočava se problem: koje simbole smjestiti na koju granu stabla. Stoga se za generiranje optimalnih kodova upotrebljavaju standardni postupci i to:

a.) Shannon-Fano metoda

b.) Huffmanova metoda

c.) Hammingova metoda

### 4.2.1. 4.2.1. SHANNON-FANO METODA

U sljedeća dva primjera pokazat ćemo dvije praktičke metode za sintezu optimalnih kodova.

Zadatak: Za vijesti što ih treba kodirati, u tablici su zadane apriorne vjerojatnosti p(ui). Shannon-Fano metodom kodiranja, kodirajte vijesti tako da dobijete optimalan kôd.

*Tablica4.1: Optimalno kodiranje (Shannon-Fano metoda)*

| $u_i$ | $p(u_i)$ | 1. | 2. | 3. | 4. | 5 | 6. | A | B | C | D | E | F | $n_i$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $u_1$ | 0,3 | | | | | | | | | | | | | |
| $u_2$ | 0,3 | | | | | | | | | | | | | |
| $u_3$ | 0,2 | | | | | | | | | | | | | |
| $u_4$ | 0,1 | | | | | | | | | | | | | |
| $u_5$ | 0,05 | | | | | | | | | | | | | |
| $u_6$ | 0,05 | | | | | | | | | | | | | |
| $u_7$ | | | | | | | | | | | | | | |
| $u_8$ | | | | | | | | | | | | | | |

Odredite binaran kôd za kodiranje odijeljenih vijesti koji će biti najbliži optimalnome.

***Algoritam:*** Shannon-Fano metoda:

- Vijesti se poredaju po padajućim vjerojatnostima.
- Takav niz razdijeli se u dvije skupine tako da su *sume vjerojatnosti pojedinih vijesti* u obje skupine što je moguće bliže 1/2.
- Svim vijestima koje su u gornjoj skupini (skupina A) pridodajemo simbol 0, što označuje *prvi znak* binarnoga kôda.
- Svim vijestima koje su u drugoj skupini (skupina A) pridodajemo simbol 1, što također označuje *prvi znak* binarnoga kôda.
- Nadalje, svaku od ostalih skupina (od A do F) opet podijelimo na podskupine, tako da je suma vjerojatnosti u svim podskupinama približno jednaka, pri tome se
  - *gornjim podskupinama* u svakoj skupini pridružuje simbol 0, što označuje *drugi znak* binarnoga kôda, a
  - *donjim podskupinama* pridružuje se simbol 1 što također ima značenje drugoga znaka.

Ovakva dioba nastavlja se tako dugo dok u svakoj podskupini ne ostane samo po jedna vijest.

### 4.2.2. 4.2.2. HUFFMANOVA METODA OPTIMALNOGA KODIRANJA ODIJELJENIH VIJESTI

***Zadatak***: Zadan je skup vijesti $u_i$ i njihove vjerojatnosti $p(u_i)$:

| vijest | $u_1$ | $u_2$ | $u_3$ | $u_4$ | $u_5$ | $u_6$ |
|---|---|---|---|---|---|---|
| vjerojatnost | 0,30 | 0,25 | 0,20 | 0,15 | 0,05 | 0,05 |

- Vijesti su poredane po padajućim vjerojatnostima. Treba pronaći optimalan kôd za kodiranje odijeljenih vijesti iz zadanoga skũpa.

***Algoritam:*** Huffmanova metoda.

- Vijesti se poredaju po padajućim vjerojatnostima.
- Posljednje dvije (one s najmanjim vjerojatnostima) združe se u jednu skupinu, a njihove vjerojatnosti se zbroje.
- Nadalje, tako dobivena skupina promatra se kao jedna vijest time da se prema iznosu vjerojatnosti stavi na odgovarajuće mjesto.
- U tako dobivenome skupu opet se združuju dvije vijesti s najmanjim vjerojatnostima i dobivena nova vijest stavlja se na određeno mjesto po veličini.
- Čitav postupak ponavlja se dok zbroj vjerojatnosti ne bude jednak 1.

Na takav način dobiva se shema kao na slici 2.4 u skripti. Da bi došli do odgovarajućega kôda za pojedine vijesti,

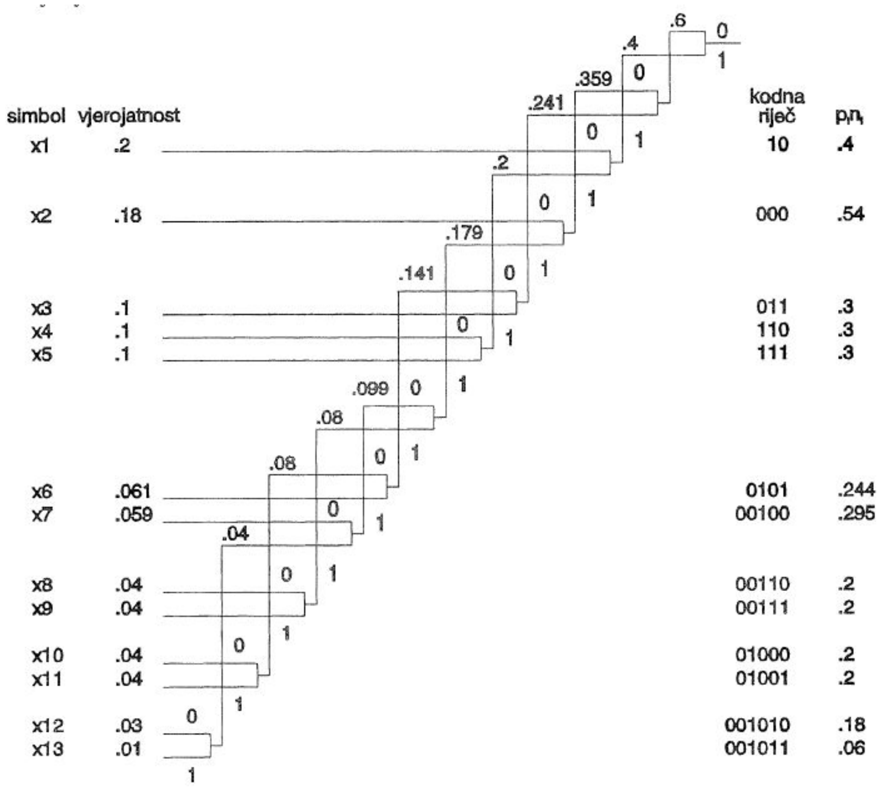
- krenimo od gornjega desnoga kuta, tako da uvijek gornju granu označimo s "1" a donju s "0".
- Broj grananja (–|) umanjen za 1 dat će nam broj binarnih simbola u kôdnoj riječi za pojedine vijesti, a
- oznake grana bit će ujedno i oznaka odgovarajućega binarnoga simbola.

Na slici 2.4 u skripti označene su kôdne riječi za pojedine vijesti.

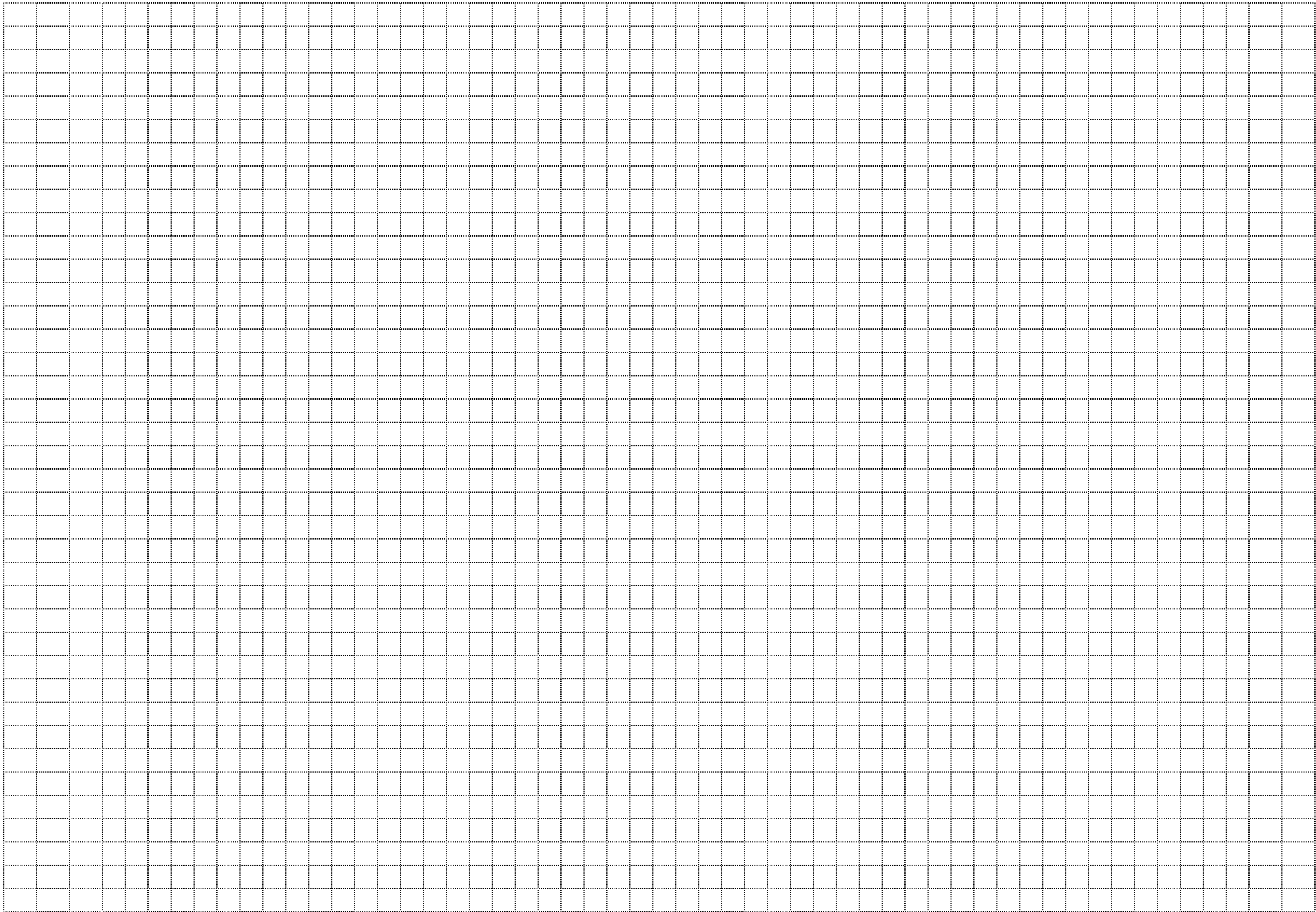
| $u_i$ | $p(u_i)$ | kôd |
|---|---|---|
| $u_1$ | | |
| $u_2$ | | |
| $u_3$ | | |
| $u_4$ | | |
| $u_5$ | | |
| $u_6$ | | |

## 2. *PRIMJER 4.2: Huffmanovo kodiranje*

Slika 4.6 ilustrira opisanu Huffmanovu metodu kao primjer kodiranja 13 poruka sa zadanim vjerojatnostima.




Slika 4.5



*Slika 4.6 Huffmanovo kodiranje.*

## 4.3. 4.3. Usporedba učinkovitosti metoda optimalnoga kodiranja

### 3. PRIMJER 4.3: Shannon-Fano i Huffmanovo kodiranje

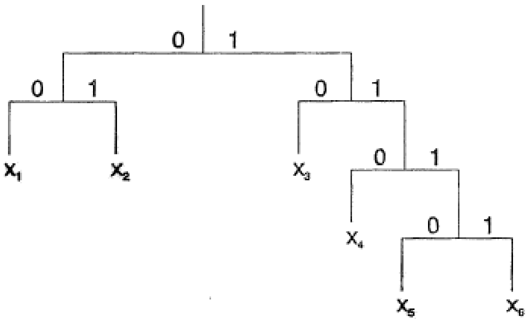
Skup od 6 poruka je određen skupom vjerojatnosti pojavljivanja kao u tablici 4.2, gdje su poruke već poredane po padajućim vjerojatnostima.

*Tablica 4.2 Shannon-Fano kodiranje*

| simbol | P($x_i$) | kodna grupa |
|---|---|---|
| $x_1$ | 0.3 | |
| $x_2$ | 0.25 | |
| $x_3$ | 0.2 | |
| $x_4$ | 0.1 | |
| $x_5$ | 0.1 | |
| $x_6$ | 0.05 | |

1. korak dijeljenja
2. korak dijeljenja
3. korak dijeljenja
4. korak dijeljenja
5. korak dijeljenja

Dijeljenjem u skupine prema opisanoj Shannon-Fano metodi dobiven je binarni kôd (slika 4.4).




*Slika 4.7 Kôdno stablo prema tablici 4.2*

Kôdna učinkovitost kôdnoga stabla na slici 4.4 je:

$$E = \frac{H}{C} = \frac{\sum_{i=1}^{6} p(x_i)ld\ p(x_i)}{\sum_{i=1}^{6} m_i p(x_i)} = \frac{2{,}366}{2{,}4} = 0{,}986$$

Ako se traži veća kôdna učinkovitost, slijedi kodiranje parova simbola. Tablica 4.2, slično tablici 4.1, daje nove simbole poredane po padajućim vjerojatnostima, te postupak dijeljenja po skupinama jednakih vjerojatnosti, kao i pripadne kôdne skupine. Dobivena učinkovitost kôda je 0.991.

*Tablica 4.3 Kodiranje po parovima.*

| simbol $x_ix_j$ | vjerojatnost $p(x_ix_j)=p(x_i)p(x_j)$ | kodna skupina | simbol $x_ix_j$ | vjerojatnost $p(x_ix_j)=p(x_i)p(x_j)$ | kodna skupina | simbol $x_ix_j$ | vjerojatnost $p(x_ix_j)=p(x_i)p(x_j)$ | kodna skupina |
|---|---|---|---|---|---|---|---|---|
| $x_1x_1$ | 0,9 | 000 | $x_5x_1$ | 0,03 | 10100 | $x_6x_2$ | 0,0125 | 111010 |
| $x_1x_2$ | 0,075 | 0010 | $x_2x_4$ | 0,025 | 10101 | $x_3x_6$ | 0,01 | 111100 |
| $x_2x_1$ | 0,075 | 0011 | $x_2x_5$ | 0,025 | 10110 | $x_4x_4$ | 0,01 | 1110110 |
| $x_2x_2$ | 0,0625 | 0100 | $x_4x_2$ | 0,025 | 10111 | $x_4x_5$ | 0,01 | 1110111 |
| $x_1x_3$ | 0,06 | 0101 | $x_5x_2$ | 0,025 | 11000 | $x_5x_4$ | 0,01 | 1111010 |
| $x_3x_1$ | 0,06 | 0110 | $x_3x_4$ | 0,02 | 11010 | $x_5x_5$ | 0,01 | 1111011 |
| $x_2x_3$ | 0,05 | 01110 | $x_3x_5$ | 0,02 | 110010 | $x_6x_3$ | 0,01 | 1111100 |
| $x_3x_2$ | 0,05 | 01111 | $x_4x_3$ | 0,02 | 110011 | $x_4x_6$ | 0,005 | 1111101 |
| $x_3x_3$ | 0,04 | 10000 | $x_5x_3$ | 0,02 | 110110 | $x_5x_6$ | 0,005 | 11111100 |
| $x_1x_4$ | 0,03 | 10001 | $x_1x_6$ | 0,015 | 110111 | $x_6x_4$ | 0,005 | 11111101 |

PRIMJER 4.3: Prirodan jezik

Slova u hrvatskome pisanome jeziku, promatra li ih neovisno, javljaju se vjerojatnostima kao u tablici 4.3. Optimalne dužine binarnoga kôda koje odgovaraju relaciji:

$$m_i = I(x_i) = -\mathrm{ld}\ p(x_i)$$

također su raspoložive u tablici 4.3. Tablica daje optimalan binaran kôd koji je određen Huffmanovom metodom. Također su za usporedbu dane vrijednosti *C* i *E* za:

1. ravnomjeran,
2. Shannon-Fano i
3. Huffmanov kôd.

*Tablica 4.3 Optimalan binarni kôd za abecedu hrvatskoga jezika.*

| znak $x_i$ | vjerojatnost $P(X_i)$ | kodna riječ | znak $x_i$ | vjerojatnost $P(X_i)$ | kodna riječ | znak $x_i$ | vjerojatnost $P(X_i)$ | kodna riječ |
|---|---|---|---|---|---|---|---|---|
| Rk | 0.17 | 010 | T | 0.0367 | 00101 | B | 0.0155 | 011101 |
| A | 0.096 | 101 | U | 0.0364 | 00010 | Z | 0.0144 | 110101 |
| E | 0.077 | 111 | D | 0.0319 | 01111 | Š | 0.0086 | 0001110 |
| O | 0.0754 | 0000 | M | 0.0313 | 10010 | Č | 0.0084 | 0001111 |
| I | 0.0742 | 0110 | V | 0.0306 | 10010 | C | 0.0067 | 0001100 |
| N | 0.0464 | 1100 | L | 0.0306 | 10000 | H | 0.0065 | 0001101 |
| J | 0.0435 | 00110 | K | 0.0298 | 10001 | Ž | 0.0052 | 1101000 |
| S | 0.042 | 00111 | P | 0.0204 | 11011 | Ć | 0.0049 | 11010010 |
| R | 0.0382 | 00100 | G | 0.0166 | 011100 | F | 0.0011 | 11010011 |

a.) $C_{\text{ravnomjeran}} = 4.754888$ $E_a = H/C = 0.8822766$
b.) $C_{\text{Shannon-Fano}} = 4.2526$ $E_b = H/C = 0.9864849$
c.) $C_{\text{Huffman}} = 4.2363$ $E_c = H/C = 0.9902808$

## 4.4. 4.4. Hammingova metoda

1. Koristeći optimalan kôd za hrvatski jezik, kodirajte vijest što sadrži Vaše:

"PREZIME IME"

| | 1 | 2 | 3 | 4 | 5 | | | 1 | 2 | 3 | 4 | 5 | 6 | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Rk[1] | 0 | 1 | 0 | | | | T | 0 | 0 | 1 | 0 | 1 | | | B | 0 | 1 | 1 | 1 | 0 | 1 | | |
| A | 1 | 0 | 1 | | | | U | 0 | 0 | 0 | 1 | 0 | | | Z | 1 | 1 | 0 | 1 | 0 | 1 | | |
| E | 1 | 1 | 1 | | | | M | 1 | 0 | 0 | 1 | 0 | | | Š | 0 | 0 | 0 | 1 | 1 | 1 | 0 | |
| O | 0 | 0 | 0 | 0 | | | D | 0 | 1 | 1 | 1 | 1 | | | Č | 0 | 0 | 0 | 1 | 1 | 1 | 1 | |
| I | 0 | 1 | 1 | 0 | | | V | 1 | 0 | 0 | 1 | 1 | | | C | 0 | 0 | 0 | 1 | 1 | 0 | 0 | |
| N | 1 | 1 | 0 | 0 | | | L | 1 | 0 | 0 | 0 | 0 | | | H | 0 | 0 | 0 | 1 | 1 | 0 | 1 | |
| J | 0 | 0 | 1 | 1 | 0 | | K | 1 | 0 | 0 | 0 | 1 | | | Ž | 1 | 1 | 0 | 1 | 0 | 0 | 0 | |
| S | 0 | 0 | 1 | 1 | 1 | | P | 1 | 1 | 0 | 1 | 1 | | | Ć | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| R | 0 | 0 | 1 | 0 | 0 | | G | 0 | 1 | 1 | 1 | 0 | 0 | | F | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |

Rješenje:

[1] Rk … razmak između riječi

## *4.5. 4.5. Unesite svoja zapažanja o vježbi*