
Zapis teksta, zvuka i videa

Današnji računari su *digitalni*. To znači da su svi podaci koji su u njima zapisani – zapisani kao nizovi brojeva. Zapisivanje tekstova, slika, zvuka i filmova u vidu brojeva zahteva pogodnu reprezentaciju koja ponekad znači i gubitak dela polaznih informacija. Kada su podaci predstavljeni u vidu brojeva, te brojeve potrebno je zapisati u računarima. Dekadni brojevni sistem koji ljudi koriste u svakodnevnom životu nije pogodan za zapis brojeva u računarima jer zahteva azbuku od 10 različitih simbola (cifara). Bilo da se radi o elektronskim, magnetnim ili optičkim komponentama, tehnologija izrade računara i medijuma za zapis podataka koristi elemente koji imaju dva diskretna stanja, što za zapis podataka daje azbuku od samo dva različita simbola. Tako, na primer, ukoliko između dve tačke postoji napon viši od određenog praga, onda se smatra da tom paru tačaka odgovara vrednost 1, a inače mu odgovara vrednost 0. Takođe, polje hard diska može biti ili namagnetisano što odgovara vrednosti 1 ili razmagnetisano što odgovara vrednosti 0. Slično, laserski zrak na površini kompakt diska „buši rupice“ kojim je određen zapis podataka pa polje koje nije izbušeno predstavlja vrednost 0, a ono koje jeste izbušeno predstavlja vrednost 1. U nastavku će biti pokazano da je azbuka od samo dva simbola dovoljna za zapisivanje svih vrsta brojeva, pa samim tim i za zapisivanje svih vrsta digitalnih podataka.

1.1 Analogni i digitalni podaci i digitalni računari

Kontinualna priroda signala. Većina podataka koje računari koriste nastaje zapisivanjem prirodnih signala. Najznačajniji primeri signala su zvuk i slika, ali se pod signalima podrazumevaju i ultrazvučni signali, EKG signali, zračenja različite vrste itd.

Signali koji nas okružuju u prirodi u većini slučajeva se prirodno mogu predstaviti neprekidnim funkcijama. Na primer, zvučni signal predstavlja promenu pritiska vazduha u zadatoj tački i to kao neprekidnu funkciju vremena. Slika se može opisati intenzitetom svetlosti određene boje (tj. određene talasne dužine) u datom vremenskom trenutku i to kao neprekidna funkcija prostora.

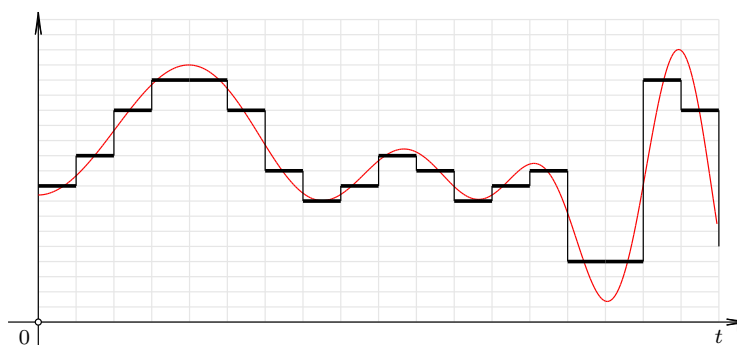
Analogni zapis. Osnovna tehnika koja se primenjuje kod analognog zapisa signala je da se kontinualne promene signala koji se zapisuje opišu kontinualnim promenama određenog svojstva medijuma na kojem se signal zapisuje. Tako, na primer, promene pritiska vazduha koji predstavlja zvučni signal direktno odgovaraju promenama nivoa namagnetisanja na magnetnoj traci na kojoj se zvuk analogno zapisuje. Količina boje na papiru direktno odgovara intenzitetu svetlosti u vremenskom trenutku kada je fotografija bila snimljena. Dakle, analogni zapis uspostavlja *analogiju* između signala koji je zapisan i određenog svojstva medijuma na kome je signal zapisan.

Osnovna prednost analogne tehnologije je da je ona obično veoma jednostavna ukoliko se zadovoljimo relativno niskim kvalitetom (još su drevni narodi mogli da naprave nekakav zapis zvuka uz pomoć jednostavne igle prikačene na trepereću membranu).

Osnovni problem analogne tehnologije je što je izrazito teško na medijumu napraviti veran zapis signala koji se zapisuje i izrazito je teško napraviti dva identična zapisa istog signala. Takođe, problem predstavlja i inherentna nestalnost medijuma, njegova promenljivost tokom vremena i podložnost spoljašnjim uticajima. S obzirom na to da varijacije medijuma direktno dovode do varijacije zapisanog signala, vremenom neizbežno dolazi do pada kvaliteta analogno zapisanog signala. Obrada analogno zapisanih signala je obično veoma komplikovana i za svaku vrstu obrade signala, potrebno je da postoji uređaj koji je specijalizovan za tu vrste obrade.

Digitalni zapis. Osnovna tehnika koja se koristi kod digitalnog zapisa podataka je da se vrednost signala izmeri u određenim vremenskim trenucima ili određenim tačkama prostora i da se onda na medijumu zapišu

izmerene vrednosti. Ovim je svaki digitalno zapisani signal predstavljen nizom brojeva koji se nazivaju *odbirci* ili *semplovi* (engl. *sample*). Svaki od brojeva predstavlja vrednost signala u jednoj tački diskretizovanog domena. S obzirom na to da izmerene vrednosti takođe pripadaju kontinualnoj skali, neophodno je izvršiti i diskretizaciju kodomena, odnosno dopustiti zapisivanje samo određenog broja nivoa različitih vrednosti.



Slika 1.1: Digitalizacija zvučnog signala

Digitalni zapis predstavlja diskretnu aproksimaciju polaznog signala. Važno pitanje je koliko često je potrebno vršiti merenje da bi se polazni kontinualni signal mogao verno rekonstruisati. Odgovor daje tvrđenje o odabiranju (tzv. Najkvist-Šenonova teorema), koje kaže da je signal dovoljno meriti dva puta češće od najviše frekvencije koja sa u njemu javlja. Na primer, pošto čovekovo uho čuje frekvencije do 20kHz, dovoljno je da frekvencija odabiranja (semplovanja) bude 40kHz. Dok je za analogne tehnologije za postizanje visokog kvaliteta zapisa potrebno imati medijume visokog kvaliteta, kvalitet reprodukcije digitalnog zapisa ne zavisi od toga kakav je kvalitet medija na kome su podaci zapisani, sve dok je medijum dovoljnog kvaliteta da se zapisani brojevi mogu razaznati. Dodatno, kvarljivost koja je inherentna za sve medije postaje nebitna. Na primer, papir vremenom žuti što uzrokuje pad kvaliteta analognih fotografija tokom vremena. Međutim, ukoliko bi papir sadržao zapis brojeva koji predstavljaju vrednosti boja u tačkama digitalno zapisane fotografije, činjenica da papir žuti ne bi predstavljala problem dok god se brojevi mogu razaznati.

Digitalni zapis omogućava kreiranje apsolutno identičnih kopija što dalje omogućava prenos podataka na daljinu. Na primer, ukoliko izvršimo fotokopiranje fotografije, napravljena fotokopija je daleko lošijeg kvaliteta od originala. Međutim, ukoliko umnožimo CD na kojem su zapisani brojevi koji čine zapis neke fotografije, kvalitet slike ostaje apsolutno isti. Ukoliko bi se dva CD-a pregledala pod mikroskopom, oni bi izgledali delimično različito, ali to ne predstavlja problem sve dok se brojevi koji su na njima zapisani mogu razaznati.

Obrada digitalno zapisanih podataka se svodi na matematičku manipulaciju brojevima i ne zahteva (za razliku od analognih podataka) korišćenje specijalizovanih mašina.

Osnovni problem implementacije digitalnog zapisa predstavlja činjenica da je neophodno imati veoma razvijenu tehnologiju da bi se uopšte stiglo do iole upotrebljivog zapisa. Na primer, izuzetno je komplikovano napraviti uređaj koji je u stanju da 40 hiljada puta izvrši merenje intenziteta zvuka. Jedna sekunda zvuka se predstavlja sa 40 hiljada brojeva, za čiji je zapis neophodna gotovo cela jedna sveska. Ovo je osnovni razlog zbog čega se digitalni zapis istorijski javio kasno. Kada se došlo do tehnološkog nivoa koji omogućava digitalni zapis, on je doneo mnoge prednosti u odnosu na analogni.

1.2 Zapis teksta

Međunarodna organizacija za standardizaciju, ISO (engl. *International Standard Organization*) definiše tekst (ili dokument) kao „informaciju namenjenu ljudskom sporazumevanju koja može biti prikazana u dvodimenzionalnom obliku. Tekst se sastoji od grafičkih elemenata kao što su karakteri, geometrijski ili fotografski elementi ili njihove kombinacije, koji čine sadržaj dokumenta“. Iako se tekst obično zamišlja kao dvodimenzioni objekat, u računarima se tekst predstavlja kao jednodimenzioni (linearni) niz karaktera koji pripadaju određenom unapred fiksnom skupu karaktera. U zapisu teksta, koriste se specijalni karakteri koji označavaju prelazak u novi red, tabulator, kraj teksta i slično.

Osnovna ideja koja omogućava zapis teksta u računarima je da se svakom karakteru pridruži određen (neoznačeni) ceo broj (a koji se interno u računarima zapisuje binarno) i to na unapred dogovoreni način. Ovi brojevi se nazivaju *kovovima karaktera* (engl. *character codes*). Tehnička ograničenja ranih računara kao i neravnomern razvoj računarstva između različitih zemalja, doveli su do toga da postoji više različitih standardnih tabela

koje dodeljuju numeričke kodove karakterima. U zavisnosti od broja bitova potrebnih za kodiranje karaktera, razlikuju se 7-bitni kodovi, 8-bitni kodovi, 16-bitni kodovi, 32-bitni kodovi, kao i kodiranja promenljive dužine koja različitim karakterima dodeljuju kodove različite dužine. Tabele koje sadrže karaktere i njima pridružene kodove obično se nazivaju *kodne strane* (engl. *code page*).

Postoji veoma jasna razlika između karaktera i njihovih grafičke reprezentacije. Elementi pisanog teksta koji najčešće predstavljaju grafičke reprezentacije pojedinih karaktera nazivaju se *glifovi* (engl. *glyph*), a skupovi glifova nazivaju se *fontovi* (engl. *font*). Korespondencija između karaktera i glifova ne mora biti jednoznačna. Naime, softver koji prikazuje tekst može više karaktera predstaviti jednim glifom (to su takozvane *ligature*, kao na primer glif za karaktere „f“ i „i“: fi), dok jedan isti karakter može biti predstavljen različitim glifovima u zavisnosti od svoje pozicije u reči. Takođe, moguće je da određeni fontovi ne sadrže glifove za određene karaktere i u tom slučaju se tekst ne prikazuje na željeni način, bez obzira što je ispravno kodiran. Fontovi koji se obično instaliraju uz operativni sistem sadrže glifove za karaktere koji su popisani na takozvanoj *WGL4* listi (*Windows Glyph List 4*) koja sadrži uglavnom karaktere korišćene u evropskim jezicima, dok je za ispravan prikaz, na primer, kineskih karaktera, potrebno instalirati dodatne fontove. Specijalnim karakterima se najčešće ne pridružuju zasebni grafički likovi.

Englesko govorno područje. Tokom razvoja računarstva, broj karaktera koje je bilo poželjno kodirati je postajao sve veći. Pošto je računarstvo u ranim fazama bilo razvijano uglavnom u zemljama engleskog govornog područja, bilo je potrebno predstaviti sledeće karaktere:

- Mala slova engleskog alfabeta: a, b, ... , z
- Velika slova engleskog alfabeta: A, B, ... , Z
- Cifre 0, 1, ..., 9
- Interpunkcijske znake: , . : ; + * - _ () [] { } ...
- Specijalne znake: kraj reda, tabulator, ...

Standardne tabele kodova ovih karaktera su se pojavile još tokom 1960-ih godina. Najrasprostranjenije od njih su:

- *EBCDIC* - IBM-ov standard, korišćen uglavnom na mejnfrejmskim računarima, pogodan za bušene kartice.
- *ASCII* - standard iz koga se razvila većina danas korišćenih standarda za zapis karaktera.

ASCII. *ASCII (American Standard Code for Information Interchange)* je standard uspostavljen 1968. godine od strane *Američkog nacionalnog instituta za standarde*, (engl. *American National Standard Institute*) koji definiše sedmobitni zapis koda svakog karaktera što daje mogućnost zapisivanja ukupno 128 različitih karaktera, pri čemu nekoliko kodova ima dozvoljeno slobodno korišćenje. *ISO* takođe delimično definiše ASCII tablicu kao deo svog standarda *ISO 646 (US)*.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	STX	SOT	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

Slika 1.2: ASCII tablica

Osnovne osobine ASCII standarda su:

- Prva 32 karaktera – od $(00)_{16}$ do $(1F)_{16}$ – su specijalni kontrolni karakteri.
- Ukupno 95 karaktera ima pridružene grafičke likove (engl. *printable characters*).

- Cifre 0-9 predstavljene su kodovima $(30)_{16}$ do $(39)_{16}$, tako da se njihov ASCII zapis jednostavno dobija dodavanjem prefiksa 011 na njihov binarni zapis.
- Kôdovi velikih i malih slova se razlikuju u samo jednom bitu u binarnoj reprezentaciji. Na primer, *A* se kodira brojem $(41)_{16}$ odnosno $(100\,0001)_2$, dok se *a* kodira brojem $(61)_{16}$ odnosno $(110\,0001)_2$. Ovo omogućava da se konverzija veličine slova u oba smera može vršiti efikasno.
- Slova su poređana u *kolacionu sekvencu*, u skladu sa engleskim alfabetom.

Različiti operativni sistemi predviđaju različito kodiranje oznake za prelazak u novi red. Tako operativni sistem Windows podrazumeva da se prelazak u novi red kodira sa dva kontrolna karaktera i to *CR* (carriage return) predstavljen kodom $(0D)_{16}$ i *LF* (line feed) predstavljen kodom $(0A)_{16}$, operativni sistem Unix i njegovi derivati (pre svega Linux) podrazumevaju da se koristi samo karakter *LF*, dok MacOS podrazumeva korišćenje samo karaktera *CR*.

Nacionalne varijante ASCII tablice i ISO 646. Tokom 1980-ih, *Jugoslovenski zavod za standarde* definisao je standard *YU-ASCII* (*YUSCII*, *JUS I.B1.002*, *JUS I.B1.003*) kao deo standarda ISO 646, tako što su kodovi koji imaju slobodno korišćenje (a koji u ASCII tabeli uobičajeno kodiraju zagrade i određene interpunkcijske znakove) dodeljeni našim dijakriticima:

YUSCII	ASCII	kôd	YUSCII	ASCII	kôd
Ž	@	$(40)_{16}$	ž	'	$(60)_{16}$
Š	[	$(5B)_{16}$	š	{	$(7B)_{16}$
Đ	\	$(5C)_{16}$	đ		$(7C)_{16}$
Č	]	$(5D)_{16}$	č	}	$(7D)_{16}$
Č	~	$(5E)_{16}$	č	~	$(7E)_{16}$

Osnovne mane YUSCII kodiranja su to što ne poštuje abecedni poredak, kao i to da su neke zagrade i važni interpunkcijski znaci izostavljeni.

8-bitna proširenja ASCII tabele. Podaci se u računaru obično zapisuju bajt po bajt. S obzirom na to da je ASCII sedmobitni standard, ASCII karakteri se zapisuju tako što se njihov sedmobitni kôd proširi vodećom nulom. Ovo znači da jednobajtni zapisi u kojima je vodeća cifra 1, tj. raspon od $(80)_{16}$ do $(FF)_{16}$ nisu iskorišćeni. Međutim, ni ovih dodatnih 128 kodova nije dovoljno da se kodiraju svi karakteri koji su potrebni za zapis tekstova na svim jezicima (ne samo na engleskom). Zbog toga je, umesto jedinstvene tabele koja bi proširivala ASCII na 256 karaktera, standardizovano nekoliko takvih tabela, pri čemu svaka od tabela sadrži karaktere potrebne za zapis određenog jezika odnosno određene grupe jezika. Praktičan problem je što postoji dvostruka standardizacija ovako kreiranih kodnih strana i to od strane ISO (International Standard Organization) i od strane značajnih korporacija, pre svega kompanije *Microsoft*.

ISO je definisao familiju 8-bitnih kodnih strana koje nose zajedničku oznaku *ISO/IEC 8859* (kodovi od $(00)_{16}$ do $(1F)_{16}$, $(7F)_{16}$ i od $(80)_{16}$ do $(9F)_{16}$ ostali su nedefinisani ovim standardom, iako se često u praksi popunjavaju određenim kontrolnim karakterima):

ISO-8859-1	Latin 1	većina zapadnoevropskih jezika
ISO-8859-2	Latin 2	centralno i istočnoevropski jezici
ISO-8859-3	Latin 3	južnoevropski jezici
ISO-8859-4	Latin 4	severnoevropski jezici
ISO-8859-5	Latin/Cyrillic	ćirilica većine slovenskih jezika
ISO-8859-6	Latin/Arabic	najčešće korišćeni arapski
ISO-8859-7	Latin/Greek	moderni grčki alfabet
ISO-8859-8	Latin/Hebrew	moderni hebrejski alfabet

Kompanija Microsoft definisala je familiju 8-bitnih strana koje se označavaju kao *Windows-125x* (ove strane se nekada nazivaju i *ANSI*). Za srpski jezik, značajne su kodne strane:

Windows-1250	centralnoevropski i istočnoevropski jezici
Windows-1251	ćirilica većine slovenskih jezika
Windows-1252	(često se neispravno naziva i ANSI) zapadnoevropski jezici

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	ı	ç	£	¤	¥	ı	§	¨	©	ª	«	¬	SHY	®	-
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	

Slika 1.3: ISO-8859-1 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	À	Á	Â	Ã	Ä	Å	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	

Slika 1.4: ISO-8859-2 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,		„	…	†	‡		‰	Š	‹	Ś	Ť	Ž	Ž
9		‘	’	“	”	•	—	—		™	š	›	ś	ť	ž	ž
A		˘	˙	Ł	ł	Ą	ą	Ś	ś	©	Ş	«	¬		®	Ž
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	

Slika 1.5: Windows-1250 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	Ě	Ě	Ě	€	Š	Š	Š	Š	Š	Š	Š	Š	SHY	Ÿ	Ÿ
B	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
C	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
D	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
E	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ	Ѧ

Slika 1.6: ISO-8859-5 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ѓ		,		„	...	†	‡		‰	Љ	<	Њ	Ќ	Ѕ	Ї
9		'	'	“	”	•	—	—		™	Ў	>	Ў	Ќ	Ѕ	Ї
A		Ÿ	Ÿ	Ј	Ѡ	Ѓ	Ѕ	Ї	Љ	Њ	Ќ	Ѕ	Ї	Ѕ	Ї	Ї
B	°	±	І	і	Ѓ	μ	¶	·	ё	№	€	»	ј	Ѕ	Ѕ	Ї
C	A	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
D	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
E	a	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
F	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	

Slika 1.7: Windows-1251 tablica

Unicode. Iako navedene kodne strane omogućavaju kodiranje tekstova koji nisu na engleskom jeziku, nije moguće, na primer, u istom tekstu koristiti i ćirilicu i latinicu. Takođe, za azijske jezike nije dovoljno 256 mesta za zapis svih karaktera. Pošto je kapacitet računara vremenom rastao, postepeno se krenulo sa standardizacijom skupova karaktera koji karaktere kodiraju sa više od jednog bajta. Kasnih 1980-ih, dve velike organizacije započele su standardizaciju tzv. univerzalnog skupa karaktera (engl. Universal Character Set — UCS). To su bili ISO, kroz standard 10646 i projekat *Unicode*, organizovan i finansiran uglavnom od strane američkih kompanija koje su se bavile proizvodnjom višejezičkog softvera (Xerox Parc, Apple, Sun Microsystems, Microsoft, ...).

ISO 10646 zamišljen je kao četvorobajtni standard. Prvih 65536 karaktera koristi se kao osnovni višejezički skup karaktera, dok je preostali prostor ostavljen kao proširenje za drevne jezike, naučnu notaciju i slično.

Unicode je za cilj imao da bude:

- univerzalan (UNiversal) — sadrži sve savremene jezike sa pismom;
- jedinstven (UNIque) — bez dupliranja karaktera - kodiraju se pisma, a ne jezici;
- uniforman (UNIform) — svaki karakter sa istim brojem bitova.

Početna verzija Unicode standarda svakom karakteru dodeljuje dvobajtni kôd (tako da kôd svakog karaktera sadrži tačno 4 heksadekadne cifre). Dakle, moguće je dodeliti kodove za ukupno $2^{16} = 65536$ različitih karaktera. S vremenom se shvatilo da dva bajta neće biti dovoljno za zapis svih karaktera koji se koriste na planeti, pa je odlučeno da se skup karaktera proširi i Unicode danas dodeljuje kodove od $(000000)_{16}$ do $(10FFFF)_{16}$ podeljenih u 17 tzv. ravni, pri čemu svaka ravan sadrži 65536 karaktera. U najčešćoj upotrebi je *osnovna višejezička ravan* (engl. *basic multilingual plane*) koja sadrži većinu danas korišćenih karaktera (uključujući i CJK — Chinese, Japanese, Korean — karaktere koji se najčešće koriste) čiji su kodovi između $(0000)_{16}$ i $(FFFF)_{16}$.

Vremenom su se pomenuta dva projekta UCS i Unicode združila i danas postoji izuzetno preklapanje između ova dva standarda.

U sledećoj tabeli je naveden raspored određenih grupa karaktera u osnovnoj višejezičkoj ravni:

0020-007E	ASCII printable
00A0-00FF	Latin-1
0100-017F	Latin extended A (osnovno proširenje latinice, sadrži sve naše dijakritike)
0180-027F	Latin extended B
...	
0370-03FF	grčki alfabet
0400-04FF	ćirilica
...	
2000-2FFF	specijalni karakteri
3000-3FFF	CJK (Chinese-Japanese-Korean) simboli
...	

Unicode standard u suštini predstavlja veliku tabelu koja svakom karakteru dodeljuje broj. Standardi koji opisuju kako se niske karaktera prevode u nizove bajtova se definišu dodatno.

UCS-2. ISO definiše UCS-2 standard koji svaki Unicode karakter osnovne višejezičke ravni jednostavno zapisuje sa odgovarajuća dva bajta.

UTF-8. Latinični tekstovi kodirani korišćenjem UCS-2 standarda sadrže veliki broj nula. Ne samo što te nule zauzimaju dosta prostora, već zbog njih softver koji je razvijen za rad sa dokumentima u ASCII formatu ne može da radi bez izmena nad dokumentima kodiranim korišćenjem UCS-2 standarda. *Unicode Transformation Format (UTF-8)* je algoritam koji svakom dvobajtnom Unicode karakteru dodeljuje određeni niz bajtova čija dužina varira od 1 do najviše 3. UTF je ASCII kompatibilan, što znači da se ASCII karakteri zapisuju pomoću jednog bajta, na standardni način. Konverzija se vrši na osnovu sledećih pravila:

raspon	binarno zapisan Unicode kôd	binarno zapisan UTF-8 kôd
0000-007F	00000000 0xxxxxxx	0xxxxxxx
0080-07FF	0000yyyy yyxxxxxx	110yyyyy 10xxxxxx
0800-FFFF	zzzzyyyy yyxxxxxx	1110zzzz 10yyyyyy 10xxxxxx

Na primer, karakter A koji se nalazi u ASCII tabeli ima Unicode kôd $(0041)_{16} = (0000\ 0000\ 0100\ 0001)_2$, pa se na osnovu prvog reda prethodne tabele u UTF-8 kodiranju zapisuje kao $(01000001)_2 = (41)_{16}$. Karakter Š ima Unicode kôd $(0160)_{16} = (0000\ 0001\ 0110\ 0000)_2$. Na njega se primenjuje drugi red prethodne tabele i dobija se da je njegov UTF-8 zapis $(1100\ 0101\ 1010\ 0000)_2$ tj. $(C5A0)_{16}$. Opisani konverzioni algoritam omogućava da se čitanje samo početka jednog bajta odredi da li je u pitanju karakter zapisan korišćenjem jednog, dva ili tri bajta.

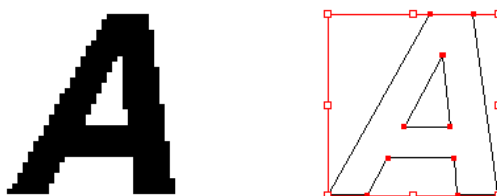
1.3 Zapis multimedijalnih sadržaja

Računari imaju sve veću ulogu u većini oblasti svakodnevnog života. Od mašina koje su pre svega služile za izvođenje vojnih i naučnih izračunavanja, računari su postali i sredstvo za kućnu zabavu (gledanje filmova, slušanje muzike), izvor informacija (internet) i nezaobilazno sredstvo u komunikaciji (elektronska pošta (engl. e-mail), časkanje (engl. chat, instant messaging), video konferencije, telefoniranje korišćenjem interneta (VoIP), ...). Nagli razvoj i proširivanje osnovne namene računara je posledica široke raspoloživosti velike količine multimedijalnih informacija (slika, zvuka, filmova, ...) koje su zapisane u digitalnom formatu. Sve ovo je posledica tehnološkog napretka koji je omogućio jednostavno i jeftino digitalizovanje signala, skladištenje velike količine digitalno zapisanih informacija, kao i njihov brz prenos i obradu.

1.3.1 Zapis slika

Slike se u računaru zapisuju koristeći *vektorski zapis*, *rasterski zapis* ili *kombinovani zapis*.

U vektorskom obliku, zapis slike sastoji se od konačnog broja geometrijskih figura (tačaka, linija, krivih, poligona), pri čemu se svaka figura predstavlja koncizno svojim koordinatama ili jednačinom u Dekartovoj ravni. Slike koje računari generišu često koriste vektorsku grafiku. Vektorski zapisane slike često zauzimaju manje prostora, dozvoljavaju uvećavanje (engl. zooming) bez gubitaka na kvalitetu prikaza i mogu se lakše preuređivati, s obzirom na to da se objekti mogu nezavisno jedan od drugoga pomerati, menjati, dodavati i uklanjati.



Slika 1.8: Primer slike u rasterskoj (levo) i vektorskoj (desno) grafici

U rasterskom zapisu, slika je predstavljena pravougaonom matricom komponenti koje se nazivaju pikseli (engl. pixel - PICture ELement). Svaki piksel je individualan i opisan jednom bojom. Raster nastaje kao rezultat digitalizacije slike. Rasterska grafika se još naziva i bitmapirana grafika. Uređaji za prikaz (monitori, projektori), kao i uređaji za digitalno snimanje slika (fotoaparati, skeneri) koriste rasterski zapis.

Modeli boja. Za predstavljanje crno-belih slika, dovoljno je boju predstaviti intenzitetom svetlosti. Različite količine svetlosti se diskretizuju u konačan broj nivoa osvetljenja čime se dobija određeni broj nijansi sive boje. Ovakav model boja se naziva *grayscale*. Ukoliko se za zapis informacije o količini svetlosti koristi 1 bajt, ukupan broj nijansi sive boje je 256. U slučaju da se slika predstavlja sa ukupno dve boje (na primer, crna i bela, kao kod skeniranog teksta nekog dokumenta) koristi se model pod nazivom *Duotone* i boja se tada predstavlja jednim bitom.

U RGB modelu boja, kombinovanjem crvene (R), zelene (G) i plave (B) komponente svetlosti reprezentuju se ostale boje. Tako se, na primer, kombinovanjem crvene i zelene boje reprezentuje žuta boja. Bela boja se reprezentuje maksimalnim vrednostima sve tri osnovne komponente, dok se crna boja reprezentuje minimalnim vrednostima osnovnih komponenti. U ovom modelu, zapis boje čini informacija o intenzitetu crvene, plave i zelene komponente. RGB model boja se koristi kod uređaja koji boje prikazuju kombinovanjem svetlosti (monitori, projektori, ...). Ukoliko se za informaciju o svakoj komponenti koristi po jedan bajt, ukupan broj bajtova za zapis informacije o boji je 3, te je moguće koristiti $2^{24} = 16777216$ različitih boja. Ovaj model se često naziva i *TrueColor* model boja.

Za razliku od aditivnog RGB modela boja, kod kojeg se bela boja dobija kombinovanjem svetlosti tri osnovne komponente, u štampi se koristi subtraktivni CMY (Cyan-Magenta-Yellow) model boje kod kojeg se boje dobijaju kombinovanjem obojenih pigmenata na belom papiru. Kako se potpuno crna teško dobija mešanjem drugih pigmenata, a i kako je njena upotreba obično daleko najčešća, obično se prilikom štampanja uz CMY pigmente koristi i crni pigment čime se dobija model CMYK.

Za obradu slika pogodni su HSL ili HSV (poznat i kao HSB) model boja. U ovom modelu, svaka boja se reprezentuje Hue (H) komponentom (koja predstavlja ton boje), Saturation (S) komponentom (koja predstavlja zasićenost boje odnosno njenu „jarkost“) i Lightness (L), Value (V) ili Brightness (B) komponentom (koja predstavlja osvetljenost).

Formati zapisa rasterskih slika. Rasterske slike su reprezentovane matricom piksela, pri čemu se za svaki piksel čuva informacija o njegovoj boji. Dimenzije ove matrice predstavljaju tzv. *apsolutnu rezoluciju* slike. Apsolutna rezolucija i model boja koji se koristi određuju broj bajtova potrebnih za zapis slike. Na primer, ukoliko je apsolutna rezolucija slike 800×600 piksela, pri čemu se koristi RGB model boje sa 3 bajta po pikselu, potrebno je ukupno $800 \cdot 600 \cdot 3B = 1440000B \approx 1.373MB$ za zapis slike. Da bi se smanjila količina informacija potrebnih za zapis slike, koriste se tehnike kompresije i to (i) kompresije bez gubitka (engl. *lossless*), i (ii) kompresije sa gubitkom (engl. *lossy*). Najčešće korišćeni formati u kojima se koristi tehnike kompresije bez gubitka danas su GIF i PNG (koji se koriste za zapis dijagrama i sličnih računarski generisanih slika), dok je najčešće korišćeni format sa gubitkom JPEG (koji se obično koristi za fotografije).

1.3.2 Zapis zvuka

Zvučni talas predstavlja oscilaciju pritiska koja se prenosi kroz vazduh ili neki drugi medijum (tečnost, čvrsto telo). Digitalizacija zvuka se vrši merenjem i zapisivanjem vazdušnog pritiska u kratkim vremenskim intervalima. Osnovni parametri koji opisuju zvučni signal su njegova amplituda (koja odgovara „glasnoći“) i frekvencija (koja odgovara „visini“). Pošto ljudsko uho obično čuje raspon frekvencija od 20Hz do 20kHz, dovoljno je koristiti frekvenciju odabiranja 40kHz, tj. dovoljno je izvršiti odabiranje oko 40 000 puta u sekundi. Na primer, AudioCD standard koji se koristi prilikom snimanja običnih audio kompakt diskova, propisuje frekvenciju odabiranja 44.1kHz. Za postizanje još većeg kvaliteta, neki standardi (miniDV, DVD, digital TV) propisuju odabiranje na frekvenciji 48kHz. Ukoliko se snima ili prenosi samo ljudski govor (na primer, u mobilnoj telefoniji), frekvencije odabiranja mogu biti i znatno manje. Drugi važan parametar digitalizacije je broj bitova kojim se zapisuje svaki pojedinačni odbirak. Najčešće se koristi 2 bajta po odbirku (16 bitova), čime se dobija mogućnost zapisa $2^{16} = 65536$ različitih nivoa amplitude.

Da bi se dobio prostorni osećaj zvuka, primenjuje se tehnika višekanalnog snimanja zvuka. U ovom slučaju, svaki kanal se nezavisno snima posebnim mikrofonom i reprodukuje na posebnom zvučniku. *Stereo* zvuk podrazumeva snimanje zvuka sa dva kanala. *Surround* sistemi podrazumevaju snimanje sa više od dva kanala (od 3 pa čak i do 10), pri čemu se često jedan poseban kanal izdvaja za specijalno snimanje niskofrekvencijskih komponenti zvuka (tzv. *bas*). Najpoznatiji takvi sistemi su 5+1 gde se koristi 5 regularnih i jedan *bas* kanal.

Kao i slika, nekomprimovan zvuk zauzima puno prostora. Na primer, jedan minut stereo zvuka snimljenog u AudioCD formatu zauzima $2 \cdot 44100 \frac{\text{sample}}{\text{sec}} \cdot 60 \text{sec} \cdot 2 \frac{B}{\text{sample}} = 10584000B \approx 10.1MB$. Zbog toga se koriste tehnike kompresije, od kojeg je danas najkorišćenija tehnika kompresije sa gubitkom MP3 (MPEG-1 Audio-Layer 3). MP3 kompresija se zasniva na tzv. psiho-akustici koja proučava koje je komponente moguće ukloniti iz zvučnog signala, a da pritom ljudsko uho ne oseti gubitak kvaliteta signala.

1.3.3 Zapis video sadržaja

Video zapis u računarstvu predstavlja kompleksan proces koji uključuje efikasno povezivanje slike i zvuka, uz primenu različitih tehnika kompresije i sinhronizacije. Video se sastoji od niza pojedinačnih *frejmova*, pri čemu je svaki frejm digitalna slika koja se kombinuje sa odgovarajućim segmentom zvučnog zapisa. *Koderi i dekoderi* (eng. codecs) igraju ključnu ulogu u ovom procesu. Koder je odgovoran za kombinovanje slike i zvuka u jedinstven *tok* (eng. stream) tokom snimanja, pri čemu se informacije komprimuju da bi se smanjila veličina fajla. Naime, i u zapisu videa, kao i zvuka javlja se redundancija podataka. Redundancija se odnosi na višak podataka koji nisu neophodni za reprodukciju, ali zauzimaju prostor i usporavaju prenos, te se uklanjanjem redundancije značajno povećava efikasnost.

Koderi (npr. *H.264*, *H.265*, *AV1*) su algoritmi koji vrše kompresiju video i audio podataka tokom snimanja. Koristi se tehnike poput vremenskog (eng. inter-frame) i prostornog (eng. intra-frame) kodiranja kako bi se smanjila redundancija. U toku ovog procesa posmatraju se uzastopni frejmovi i analiziraju sličnosti između njih. Koderi koriste različite algoritme, poput diskretne kosinusne transformacije (eng. Discrete Cosine Transform, DCT) i kompenzacije pokreta (eng. Motion Compensation) za smanjenje veličine podataka. Glavna ideja iza DCT je da se podaci rastave na različite frekvencije, pri čemu je većina u niskim frekvencijama. Niske frekvencije se mogu zadržati, dok se visoke frekvencije, koje predstavljaju manje značajne detalje, mogu delimično ili potpuno ukloniti, čime se smanjuje veličina podataka bez značajnog gubitka vizuelnog kvaliteta. Kod tehnike kompenzacije pokreta glavna ideja je da se pokretni elementi u video zapisu efikasno predstavljaju pomoću modela predviđanja. Umesto da se kompletan frejm kodira iznova, kodira se samo razlika između trenutnog i prethodnog frejma, koristeći vektore pokreta za opisivanje kretanja objekata. *H.265*, poznat i kao *HEVC*, koristi naprednije tehnike predviđanja i particionisanja kako bi omogućio efikasniju kompresiju u odnosu na stariji *H.264*, dok *AV1*, kao projekat otvorenog koda, donosi još bolju kompresiju.

Prilikom reprodukcije, dekodirer razdvaja podatke i rekonstruiše originalne frejmove slike zajedno sa odgovarajućim audio segmentima. Koriste se sofisticirani algoritmi za dekompresiju i rekonstrukciju podataka, odvajanjem frejmova slike od odgovarajućih audio segmenata i pritom osiguravajući da se slika i zvuk *sinhronizovano* i precizno prikazuju korisniku. Sinhronizacija između zvuka i slike postiže se pomoću *vremenskih oznaka* (eng. timestamp) koji osiguravaju da se svaki deo zvučnog zapisa reprodukuje u tačno definisanom trenutku tokom prikazivanja određenog frejma. Ovaj proces je ključan za postizanje prirodnog osećaja pokreta i zvuka, jer svaki nesklad može izazvati percepcijski disonant kod korisnika. Dekoderi koriste tehnike kao što su kompenzacija kretanja zasnovana na blokovima (eng. Block-based Motion Compensation) i inverzna kosinusna transformacija (eng. Inverse Discrete Cosine Transform, IDCT) kako bi povratili originalni kvalitet slike i zvuka.

Moderni video formati kao što su *MP4*, *MKV*, ili *MOV* koriste tzv. kontejnere koji integrišu video i audio tokove, ali i druge podatke, kao što su prevodi (eng. title) ili metapodaci. Kontejneri služe kao *omoti* koji omogućavaju simultano čitanje i reprodukciju svih ovih tokova na uređajima za reprodukciju, osiguravajući sinhronizaciju između zvučnih i vizuelnih elemenata. Na ovaj način, tokom reprodukcije, dekodirer u realnom vremenu analizira, odvajajući i dekodirajući frejmove videa i odgovarajuće segmente zvučnog zapisa, koristeći sofisticirane algoritme za predviđanje i rekonstrukciju kako bi održao preciznu sinhronizaciju, čak i u slučajevima gubitka podataka ili varijabilnih brzina prenosa. *MP4* je široko podržan format i često se koristi za uživo puštanje sadržaja zbog balansa između kvaliteta i veličine fajla. *MKV* pruža veću fleksibilnost i može integrisati više tokova (npr. više audio zapisa ili titlova), dok *MOV* format, razvijen od strane kompanije Apple, nudi visok nivo kvaliteta i često se koristi u profesionalnim okruženjima za uređivanje videa.

Jedan od ključnih izazova u zapisu videa je balans između kvaliteta i efikasnosti. Pojmovi kao što su *HD*, *Full HD*, i *4K* označavaju različite rezolucije video zapisa koje direktno utiču na kvalitet slike. *HD* (*High Definition*) se odnosi na rezoluciju od 1280×720 piksela, dok *Full HD* ima rezoluciju od 1920×1080 piksela, što se često označava kao *1080p* na platformama poput YouTube servisa. Ovo *p* označava progresivno skeniranje, što znači da se svaki frejm prikazuje u celosti, što doprinosi boljem kvalitetu slike. U manjim formatima ili u slučaju upletenog (eng. interlaced) skeniranja, frejmovi se ne prikazuju svi odjednom, već se deli na polja, gde se prvo prikazuje polovina slike, a zatim druga polovina, što može dovesti do smanjenja kvaliteta slike ili povećanja *treperenja*. Više rezolucije, kao što su *4K* (3840×2160 piksela), pružaju još veću oštrinu i detalje, ali zahtevaju i veći protok podataka i veću memoriju za skladištenje.

Brzina bita (brzina protoka, eng. bitrate), izražen u megabitima po sekundi (Mbps), označava količinu podataka koja se prenosi ili obrađuje u jedinici vremena tokom reprodukcije videa. Viša brzina bita rezultira boljim kvalitetom slike, ali takođe zahteva veći kapacitet za skladištenje i širu mrežnu propusnost, dok niži brzina bita može uzrokovati gubitak detalja i pojavu vizuelnih nepravilnosti u slici. Brzina bita je u korelaciji sa rezolucijom i korišćenim kodekom – veće rezolucije, kao što su *4K*, obično zahtevaju veću brzinu bita za očuvanje kvaliteta. Na primer, za *Full HD* (*1080p*) video često se preporučuje brzina bita između 8-12 Mbps, dok za *4K* rezoluciju je potrebno da brzina bita bude između 35-45 Mbps kako bi se očuvao visok nivo detalja i kvaliteta slike. Takođe, moderniji kodeci, kao što su *H.265* i *AV1*, omogućavaju bolju kompresiju i kvalitet slike pri nižoj brzini bita u odnosu na starije kodeke, kao što je *H.264*. Na primer, *H.264* za *Full HD* (*1080p*)

kvalitet može zahtevati brzinu bita od 8-12 Mbps, dok *H.265* može postići sličan kvalitet pri 4-6 Mbps, a *AV1* može još više smanjiti potrebnu brzinu bita, omogućavajući sličan kvalitet slike pri brzina bita od 3-5 Mbps za *Full HD* (1080p), što ga čini pogodnim za reprodukovanje sadržaja preko mreža sa ograničenom propusnošću.

Kompresija sa gubicima (eng. lossy compression) koristi psihovizuelne modele kako bi eliminisala informacije koje ljudsko oko manje primećuje, čime se drastično smanjuje veličina fajla. Na primer, visoke frekvencije boja i svetlosti koje oko teško registruje često se odstranjuju iz zapisa, što omogućava efikasniju upotrebu memorije. Pored toga, adaptivne metode kodiranja kao što je kvantizacija omogućavaju dinamičko prilagođavanje nivoa detalja u zavisnosti od kompleksnosti scene, čime se optimizuje kompresija u realnom vremenu. Ove tehnike zajedno omogućavaju da moderni video zapisi zadrže visok nivo vizuelnog kvaliteta uz relativno mali prostor za skladištenje, što je od ključnog značaja za primenu u prenosu podataka i skladištenju na savremenim uređajima.

Pitanja i zadaci za vežbu

Pitanje 1.1. *Kako su u digitalnom računaru zapisani svi podaci? Koliko se cifara obično koristi za njihov zapis?*

Pitanje 1.2. *Koje su osnovne prednosti digitalnog zapisa podataka u odnosu na analogni? Koji su osnovni problemi u korišćenju digitalnog zapisa podataka?*

Pitanje 1.3. *Šta je to glif, a šta font? Da li je jednoznačna veza između karaktera i glifova? Navesti primere.*

Pitanje 1.4. *Koliko bitova koristi ASCII standard? Šta čini prva 32 karaktera ASCII tabele? Kako se određuje binarni zapis karaktera koji predstavljaju cifre?*

Pitanje 1.5. *Navesti barem dve jednobajtna kodne strane koje sadrže ćirilčne karaktere.*

Pitanje 1.6. *Nabrojati bar tri kodne sheme u kojima može da se zapiše reč računarstvo.*

Pitanje 1.7. *Koliko bitova koristi ASCII tabela karaktera, koliko YUSCII tabela, koliko ISO-8859-1, a koliko osnovna Unicode ravan? Koliko različitih karaktera ima u ovim tabelama?*

Pitanje 1.8. *Koja kodiranja teksta je moguće koristiti ukoliko se u okviru istog dokumenta želi zapisivanje teksta koji sadrži jedan pasus na srpskom (pisan latinicom), jedan pasus na nemačkom i jedan pasus na ruskom (pisan ćirilicom)?*

Pitanje 1.9. *U čemu je razlika između Unicode i UTF-8 kodiranja?*

Pitanje 1.10. *Prilikom prikazivanja filma, neki program prikazuje titlove tipa "računari æ biti...". Objasniti u čemu je problem.*

Pitanje 1.11. *Šta je to piksel? Šta je to sengl?*

Zadatak 1.1. *Odrediti broj bitova neophodan za kodiranje 30 različitih karaktera.*

✓

Zadatak 1.2. *Znajući da je dekadni kôd za karakter A 65, navesti kodirani zapis reči FAKULTET ASCII kodovima u heksadekadnom zapisu. Dekodirati sledeću reč zapisanu u ASCII kodu heksadekadno: 44 49 53 4B 52 45 54 4E 45.*

✓

Zadatak 1.3. *Korišćenjem ASCII tablice odrediti kodove kojima se zapisuje tekst: "Programiranje 1". Kôdove zapisati heksadekadno, oktavno, dekadno i binarno. Šta je sa kodiranjem teksta Matematički fakultet?*

✓

Zadatak 1.4. *Za reči računarstvo, informatika, navesti da li ih je moguće kodirati narednim metodima i, ako jeste, koliko bajtova zauzimaju:*

- (a) ASCII (b) Windows-1250 (c) ISO-8859-5
(d) ISO-8859-2 (e) Unicode (UCS-2) (f) UTF-8

✓

Zadatak 1.5. *Odrediti (heksadekadno predstavljene) kodove kojima se zapisuje tekst kružić u UCS-2 i UTF-8 kodiranjima. Rezultat proveriti korišćenjem HEX editora.*

✓

Zadatak 1.6. *HEX editori su programi koji omogućavaju direktno pregledanje, kreiranje i ažuriranje bajtova koji sačinjavaju sadržaja datoteka. Korišćenjem HEX editora pregledati sadržaj nekoliko datoteka različite vrste (tekstualnih, izvršivih programa, slika, zvučnih zapisa, video zapisa, ...).*

Zadatak 1.7. Uz pomoć omiljenog editora teksta (ili nekog naprednijeg, ukoliko editor nema tražene mogućnosti) kreirati datoteku koja sadrži listu imena 10 vaših najomiljenijih filmova (pisano latinicom uz ispravno korišćenje dijakritika). Datoteka treba da bude kodirana kodiranjem:

(a) Windows-1250 (b) ISO-8859-2 (c) Unicode (UCS-2) (d) UTF-8

Otvoriti zatim datoteku iz nekog pregledača Veba i proučiti šta se dešava kada se menja kodiranje koje pregledač koristi prilikom tumačenja datoteke (obično meni **View->Character encoding**). Objasniti i unapred pokušati predvideti ishod (uz pomoć odgovarajućih tabela koje prikazuju kodne rasporede).

Zadatak 1.8. Za datu datoteku kodiranu UTF-8 kodiranjem, korišćenjem editora teksta ili nekog od specijalizovanih alata (na primer, `iconv`) rekodirati ovu datoteku u ISO-8859-2. Eksperimentisati i sa drugim kodiranjima.

Elektronska verzija (2024)