

Увод у рачунарство 1

Александар Картељ
kartelj@matf.bg.ac.rs

Запис података

- Подаци могу бити:
 - Нумерички
 - Цели бројеви
 - Бинарно кодирани декадни бројеви
 - Реални бројеви у фиксном зарезу
 - Реални бројеви у покретном зарезу
 - Знаковни (алфанумерички)
- Груписањем ових података даље можемо представљати:
 - Текст
 - Сlike
 - Звукове
 - Видео садржај, ...

Мерне јединице података

Број битова	Хексад. цифара	Број байтова	Назив	Хексадекадна адреса поравнања првог байта
1	1/4	1/8	Бит	Није директно адресив
4	1	1/2	Полубајт	Није директно адресив
8	2	1	Бајт	Било која адреса
16	4	2	Полурећ	0, 2, 4, 6, 8, А, С, Е, ...
32	8	4	Рећ	0, 4, 8, С, ...
64	16	8	Двоstrука рећ	0, 8, ...

Бројевни системи

1. **Непозициони** – цифра има исту вредност без обзира на позицију у запису броја
2. **Позициони** – вредност цифре зависи од њене позиције у запису броја

0	1	2	3	4	5	6	7	8	9
·	।	ॢ	ॣ	।	॥	०	१	२	३
I	II	III	IV	V	VI	VII	VIII	IX	X
०	१	२	३	४	५	६	७	८	९
൦	൧	൨	൩	൪	൫	൬	൭	൮	൯
၀	၁	၂	၃	၄	၅	၆	၇	၈	၉
〇	一	二	三	四	五	六	七	八	九

Позициони бројевни систем

- Број је представљен путем ниске цифара
- Цифра на позицији i има придружену тежину N^i
- N се назива основа (база) бројевног система
- Општа форма броја у таквом систему се записује:

$$(\dots a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3} \dots)_N$$

- Опсег вредности за било коју цифру a_i је цео број из опсега $0 \leq a_i < N$

Вредност броја

- Вредност записаног броја се најчешће рачуна према формули:

$$\left(\dots a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3} \dots \right)_N = \\ \dots + a_3 N^3 + a_2 N^2 + a_0 N^0 + a_{-1} N^{-1} + \dots$$

Неки битнији позициони системи

- Унарни систем – цифра {1}
 - $(11111)_1 = (5)_{10}$
- Бинарни систем – цифре {0, 1}
 - Користи се у савременим дигиталним рачунарима
 - Пример: $(01110)_2 = 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (8)_{10} + (4)_{10} + (2)_{10} = (14)_{10}$
- Троични систем – цифре {0, 1, 2}
- Балансирани троични систем – цифре {-1, 0, 1}
 - Пример: $(110-11)_{bt} = 1 \times 3^4 + 1 \times 3^3 + 0 \times 3^2 - 1 \times 3^1 + 1 \times 3^0 = (81)_{10} + (27)_{10} - (3)_{10} + (1)_{10} = (106)_{10}$
- Октални систем – цифре {0, 1, 2, 3, 4, 5, 6, 7}

Неки битнији позициони системи (2)

- Декадни систем користи цифре {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
 - Људи највише користе овај систем, зашто?
- Хексадекадни систем користи {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}
 - Пример: $(CDE92)_{16} = 12 \times 16^4 + 13 \times 16^3 + 14 \times 16^2 + 9 \times 16^1 + 2 \times 16^0$
 $= (786432)_{10} + (53248)_{10} + (3584)_{10} + (144)_{10} + (2)_{10} = (843410)_{10}$
 - Такође битан у рачунарству
- Систем са негативном основом – основа r може бити негативна
 - Какве су последице, нпр. Израчунати декадну вредност броја $(123)_{-5}$?
- Систем са променљивом основом – вредности позиција се задају
 - Нпр. Израчунати вредност броја 1234 у основи (1,2,1,4)?

Законитости позиционих система

- Повећањем основе бројевног система смањује се дужина записа
- У свим бројевним системима се број који представља основу записује као 10
 - Пробајте нпр. за основе 2, 4, -13, ...

Запис разломљених бројева

- Запис у којем тачка основе раздваја цео део од разломљеног
- Запис помоћу n цифара:
 - Где је са m цифара записан разломљени део
 - A са $m-n$ цифара цео део
 - Грешка уколико је m веће од n
- Два типа записа:
 - Фиксни зарез – тачка (зарез) увек на истом месту
 - Покретни зарез – тачка (зарез) се „шета“
 - Број се записује као уређен пар (F,E) где су и F и E представљени у фиксном зарезу
 - Вредност се добија по формули $F \times N^E$

Разломљени бројеви – фиксни зарез

Broj	Format zapisa			
	7.4	5.3	6.1	8.0
	---.----	---.---	-----.-	-----.
$(1.3543)_{10}$	□□1.3543	□1.354	□□□□1.3	□□□□□□□□1.
$(12.7)_{10}$	□12.7000	12.700	□□□12.7	□□□□□□□12.
$(1347)_{10}$	*****	*****	□1347.0	□□□□1347.
$(123.456)_8$	123.4560	*****	□□123.4	□□□□□□123.
$(AB.1)_{16}$	□AB.1000	AB.100	□□□AB.1	□□□□□□□AB.
$(1011.1101)_2$	*****	*****	□1011.1	□□□□1011.
$(0.1101)_2$	□□0.1101	□0.110	□□□□0.1	□□□□□□□□0.

Разломљени бројеви – покретни зарез

Broj	Neki mogući zapisi		
	Zapis 1	Zapis 2	Normalizovan zapis
$(13.543)_{10}$	$(13.543, 0)$	$(0.13543, +2)$	$(1.3543, +1)$
$(12.7)_{10}$	$(127000.0, -4)$	$(0.00127, +4)$	$(1.27, 1)$
$(5347)_{10}$	$(53470., -1)$	$(0.005347, +6)$	$(5.347, +3)$
$(123.22)_4$	$(12322.000, -2)$	$(0.012322, +10)$	$(1.2322, +2)$
$(AB.1)_{16}$	$(AB10., -2)$	$(0.000AB1, +5)$	$(A.B1, +1)$
$(1011.1101)_2$	$(10111101, -100)$	$(10.111101, +10)$	$(1.0111101, +11)$
$(0.1101)_2$	$(110.10, -11)$	$(1101.0, -100)$	$(1.101, -1)$

Превођење бројева у друге основе

- Задатак је број дат у основи N :

$$(X)_N = x_n x_{n-1} \dots x_0 \cdot x_{-1} \dots x_{-m}$$

Превести у број са основом M :

$$(X)_N = y_p y_{p-1} \dots y_0 \cdot y_{-1} \dots y_{-q}$$

- Јасно је да ће се током тог поступка и број цифара, као и њихове ознаке највероватније променити
- Одвојено се преводи цео и разломљени део

Превођење целих бројева

- Поступак:
 - Вршимо целобројно дељење броја $(X)_N$ бројем M док год је то могуће
 - Током тог поступка памтимо у сваком кораку остатак при дељењу
 - На крају прочитамо списак остатака при дељењу у обрнутом редоследу

i	0	1	2	...	p
X_i	X_0	X_1	X_2	...	X_p
y_i	y_0	y_1	y_2	...	y_p

← smer čitanja cifara

X_{i+1} – celobrojni deo količnika X_i/M

y_i – ostatak pri ovom deljenju

Превођење целих бројева (2)

1. $94_{10} \rightarrow (1011110)_2$

i	0	1	2	3	4	5	6
X_i	94	47	23	11	5	2	1
y_i	0	1	1	1	1	0	1

2. $AB_{16} \rightarrow (10101011)_2$

i	0	1	2	3	4	5	6	7
X_i	AB	55	2A	15	A	5	2	1
y_i	1	1	0	1	0	1	0	1

3. $(101110)_2 \rightarrow (2E)_{16}$

i	0	1	2
X_i	101110	10	0
y_i	1110	10	0

Превођење разломљеног дела

- Поступак:
 - Вршимо множење броја $(X)_N$ бројем M и сваки пут уклањамо цео део уколико постоји
 - Притом памтимо разломљени део за следећи корак
 - Када се разломљени део изједначи са 0 , онда смо добили тачан број
 - Алтернативно, разломљени део не мора да се изједначи са 0 , онда смо добили само апроксимацију броја

Превођење разломљеног дела (2)

1. $(0,84375)_{10} \rightarrow (0,11011)_2$

i	0	1	2	3	4	5
X_{-i}	0,84375	0,68750	0,3750	0,750	0,50	0,00
y_{-i}	0	1	1	0	1	1

2. $(0,4)_{10} \approx (0,011001100\dots)_2$

i	0	1	2	3	4	5	6	7	8	9
X_{-i}	0,4	0,8	0,6	0,2	0,4	0,8	0,6	0,2	0,4	0,8
y_{-i}	0	0	1	1	0	0	1	1	0	0

3. $(0,4)_{10} \approx (0,1212\dots)_4$

i	0	1	2	3	4
X_{-i}	0,4	0,6	0,4	0,6	0,4
y_{-i}	0	1	2	1	2

Олакшано превођење

- У случајевима када је $N = M^s$, $s > 1$, може се користити олакшано превођење
- Тада се број $(X)_N$ може поделити на сегменте од по s цифара, и сваки сегмент превести независно
- Ово се односи и на цео и на разломљени део
- Олакшано се може преводити и у супротном смеру $M \rightarrow N$

Олакшано превођење (2)

- Превођење из бинарног у хексадекадни и обратно

0000 = 0	0100 = 4	1000 = 8	1100 = C
0001 = 1	0101 = 5	1001 = 9	1101 = D
0010 = 2	0110 = 6	1010 = A	1110 = E
0011 = 3	0111 = 7	1011 = B	1111 = F

$(ABC,DE)_{16} \longrightarrow (1010|1011|1100,1101|1110)_2$
 $\longrightarrow (101010111100,11011110)_2$

Олакшано превођење (3)

- На индиректан начин се, дакле, може превести из у хексадекадног у октални преко бинарног

$$\begin{aligned} (101010111100,11011110)_2 &\longrightarrow \\ (101|010|111|100,110|111|100)_2 &\longrightarrow (5274,674)_8 \end{aligned}$$

Запис означених бројева

- Шта је означен, а шта неозначен број?
- Општи принцип репрезентације:
 - Ако је $(X)_N = x_{n-2} \dots x_0$ неозначен број у систему основе N тада се означен представља помоћу додатне цифре на месту највеће тежине:
 $Y = +/- X = y_{n-1} y_{n-2} \dots y_0$
- Познати начини записа су:
 1. Запис са знаком и апсолутном вредности
 2. Запис са комплементом броја
 3. Запис са увећањем

Запис знака и апсолутне вредности

- Број се записује у форми:
 - $Y = +/- X = y_{n-1}y_{n-2}\dots y_0$ где је са y_{n-1} представљен знак броја а са остатком цифара његова вредност
 - Знак броја је позитиван ако $y_{n-1} = 0$, а негативан у случају $y_{n-1} = N-1$
 - N је основа система
- Пример:
 - $(001101)_2 = (13)_{10}$
 - $(101101)_2 = (-13)_{10}$
 - $(012)_3 = (5)_{10}$
 - $(212)_3 = (-5)_{10}$

Запис знака и апсолутне вредности (2)

- Колики је опсег вредности за бројеве дужине m ?
- Како се мења знак?
- Да ли сваки број има јединствену репрезентацију?

Запис помоћу комплемента

- Позитивни бројеви се записују исто као у запису знак и апсолутна вредност
- Најчешће се користе два типа комплемента:
 - Непотпуни (или 1-комплемент)
 - Потпуни (2-комплемент)
- У непотпуном се свака цифра замени својом комплементарном
 - Нпр. у бинарном систему се $0 \rightarrow 1$, а $1 \rightarrow 0$
 - Негација броја $(001101)_2$ је дакле $(110010)_2$
- У потпуном комплементу се на непотпуни комплемент и са њим се потом сабере вредност 1:
 - Негација броја $(001101)_2 \rightarrow (110010)_2 + (1)_2 = (110011)_2$

Запис помоћу комплемента (2)

- Да ли непотпуни комплемент има јединствену репрезентацију за сваки број?
- Да ли потпуни комплемент има јединствену репрезентацију за сваки број?
- Видећемо код аритметике да потпуни комплемент има одређене предности...

Запис уз додавање увећања

- Специјалан случај записа помоћу комплемента у коме се на вредност комплемента дода константа k
- Вредност k се назива увећање или вишак
- Неки вид уопштења потпуног комплемента

Поређење записа означених бројева

Broj	Znak i apsolutna vrednost	$N - 1$ -vi komplement	N -ti komplement	Višak 4
$(+127)_{10}$	0127	0127	0127	0131
$(-127)_{10}$	9127	9872	9873	9877
$(+64)_8$	064	064	064	070
$(-64)_8$	764	713	714	720
$(+AB)_{16}$	0AB	0AB	0AB	0AF
$(-AB)_{16}$	FAB	F54	F55	F59
$(+101)_2$	0101	0101	0101	01001
$(-101)_2$	1101	1010	1011	11111

Запис знаковних података и текста

- Бројевима (кодovima) се додељују знакови
- Примери неких познатијих знаковних кодова:
 - ASCII – 7 битни код, 8. бит се користи за контролу парности
 - EBCDIC – 8 битни код
 - ISO-8 – 8 битни код, првих 127 позиција као у ASCII
 - UNICODE – 16-битни код
 - UTF-8 – кодирање UNICODE карактера променљивом дужином (од 1-4 бајта)

ASCII код

ASCII value	Character	Control character	ASCII value	Character	ASCII value	Character	ASCII value	Character
000	(null)	NUL	032	(space)	064	@	096	
001	☺	SOH	033	!	065	A	097	a
002	☻	STX	034	"	066	B	098	b
003	♥	ETX	035	#	067	C	099	c
004	♦	EOT	036	\$	068	D	100	d
005	♣	ENQ	037	%	069	E	101	e
006	♠	ACK	038	&	070	F	102	f
007	(beep)	BEL	039	'	071	G	103	g
008	■	BS	040	(072	H	104	h
009	(tab)	HT	041)	073	I	105	i
010	(line feed)	LF	042	*	074	J	106	j
011	(home)	VT	043	+	075	K	107	k
012	(form feed)	FF	044	,	076	L	108	l
013	(carriage return)	CR	045	-	077	M	109	m
014	♪	SO	046	.	078	N	110	n
015	☼	SI	047	/	079	O	111	o
016	▼	DLE	048	0	080	P	112	p
017	▲	DC1	049	1	081	Q	113	q
018	↕	DC2	050	2	082	R	114	r
019	!!	DC3	051	3	083	S	115	s
020	π	DC4	052	4	084	T	116	t
021	\$	NAK	053	5	085	U	117	u
022	▬	SYN	054	6	086	V	118	v
023	↕	ETB	055	7	087	W	119	w
024	↑	CAN	056	8	088	X	120	x
025	↓	EM	057	9	089	Y	121	y
026	→	SUB	058	:	090	Z	122	z
027	←	ESC	059	;	091	[123	{
028	(cursor right)	FS	060	<	092	\	124	
029	(cursor left)	GS	061	=	093]	125	}
030	(cursor up)	RS	062	>	094	^	126	~
031	(cursor down)	US	063	?	095	-	127	☐

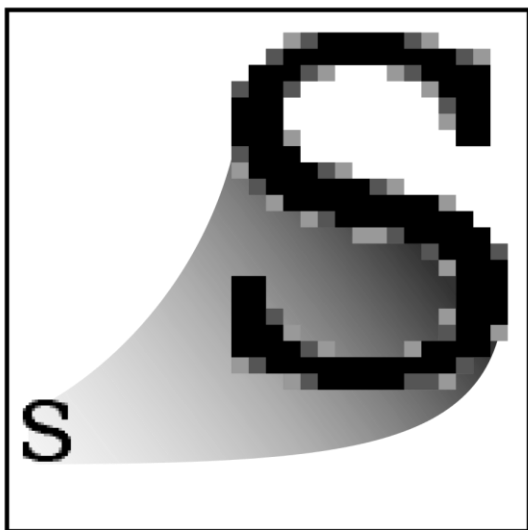
Copyright 1999, JimPrice.Com Copyright 1982, Leading Edge Computer Products, Inc.

Запис слика

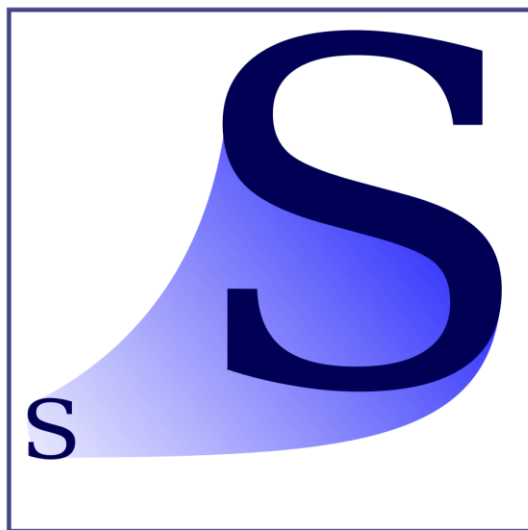
- Два основна механизма:
 - Растерски запис – матрица тачака (пиксела)
 - Векторски запис – описивање објеката (фигура)
- Може се користити и комбиновани запис
- Модели боја:
 - Црно-беле слике – колико података за сваки пиксел?
 - Сиве нијансе (grayscale)
 - Опис свим бојама:
 - RGB (red, green, blue) модел за мониторе – 8 битова, 16 битова, 32 бита, ...
 - CMYK (cyan, magenta, yellow, k-црна) модел за штампаче
 - HSV (hue, saturation, value) модел за обраду слика

Запис слика (2)

- Израчунати потребну количину меморије за слику у резолуцији 800 x 600 која користи RGB модел и по 8 битова за сваку боју?



Raster
.jpeg .gif .png



Vector
.svg

Запис звука

- Дигитализација (памћење) звука се врши мерењем и записивањем ваздушног притиска у кратким интервалима
- Величина интервала је обрнуто сразмерна брзини узорковања (sampling rate)
- Људско ухо чује распон фреквенција од 20Hz до 20KHz
- Према теорији информација потребно је највише дуплот толико узорковати, у пракси обично 44KHz
- Обично се памти 2 бајта по узорку, што омогућава око 65K амплитуда
- Дакле, минут аудио садржаја би могао да заузима:
 $44000 \text{ узорака/сек} \times 60 \text{ сек} \times 2 \text{ В/узорку} = 528000\text{В} \sim 5\text{МВ}$