



Številski sistemi in kode



Številski sistemi

Najpreprostejše štetje – zarezze (od 6000 pr.n.št.)



Evropa



Vzhodna Azija

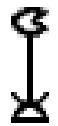





Južna Amerika



Številski sistemi

Egipčanski sistem (od 3000 pr.n.št.)

1	10	100	1000	10^4	10^5	10^6
I	∩	∩				
črtica	jarem	vrv	vodna lilija	prst roke	žaba	mož se čudi



Številski sistemi

Egipčanski sistem (od 3000 pr.n.št.)

$$= 3244$$

$$= 21237$$



Številski sistemi

Babilonski sistem (od 1900 pr.n.št.)

1	𐎶	11	𐎠𐎶	21	𐎠𐎠𐎶	31	𐎠𐎠𐎠𐎶	41	𐎠𐎠𐎠𐎶	51	𐎠𐎠𐎠𐎶
2	𐎶𐎶	12	𐎠𐎶𐎶	22	𐎠𐎠𐎶𐎶	32	𐎠𐎠𐎠𐎶𐎶	42	𐎠𐎠𐎠𐎶𐎶	52	𐎠𐎠𐎠𐎶𐎶
3	𐎶𐎶𐎶	13	𐎠𐎶𐎶𐎶	23	𐎠𐎠𐎶𐎶𐎶	33	𐎠𐎠𐎠𐎶𐎶𐎶	43	𐎠𐎠𐎠𐎶𐎶𐎶	53	𐎠𐎠𐎠𐎶𐎶𐎶
4	𐎶𐎶𐎶𐎶	14	𐎠𐎶𐎶𐎶𐎶	24	𐎠𐎠𐎶𐎶𐎶𐎶	34	𐎠𐎠𐎠𐎶𐎶𐎶𐎶	44	𐎠𐎠𐎠𐎶𐎶𐎶𐎶	54	𐎠𐎠𐎠𐎶𐎶𐎶𐎶
5	𐎶𐎶𐎶𐎶𐎶	15	𐎠𐎶𐎶𐎶𐎶𐎶	25	𐎠𐎠𐎶𐎶𐎶𐎶𐎶	35	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶	45	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶	55	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶
6	𐎶𐎶𐎶𐎶𐎶𐎶	16	𐎠𐎶𐎶𐎶𐎶𐎶𐎶	26	𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶	36	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶	46	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶	56	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶
7	𐎶𐎶𐎶𐎶𐎶𐎶𐎶	17	𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶	27	𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶	37	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶	47	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶	57	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶
8	𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	18	𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	28	𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	38	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	48	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	58	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶
9	𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	19	𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	29	𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	39	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	49	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶	59	𐎠𐎠𐎠𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶𐎶
10	𐎠	20	𐎠𐎠	30	𐎠𐎠𐎠	40	𐎠𐎠𐎠𐎠	50	𐎠𐎠𐎠𐎠		



Številski sistemi

Babilonski sistem (od 1900 pr.n.št.)



$$= 55 * 60^2 + 11 * 60^1 + 5 * 60^0 = 198665$$



Številski sistemi

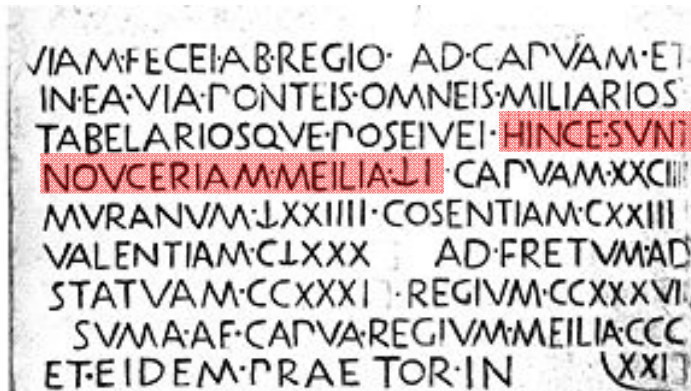
Rimski sistem (od 250 pr.n.št.)

1	5	10	50	100	500	1000
I	V	X	(⊥) L	(∩IC) C	(P) D	(Φ) M



Številski sistemi

Rimski sistem (od 250 pr.n.št.)



OD TUKAJ DO NUCERIE JE MILJ 51,
DO KAPUE 84, DO MORANUMA 74,
DO COSENTIE 123, DO VALENTIE
180, ...





Številski sistemi

Indijsko – arabski sistem = desetiški z ničlo (od 300 pr.n.št.)

European	0	1	2	3	4	5	6	7	8	9
Arabic-Indic	٠	١	٢	٣	٤	٥	٦	٧	٨	٩
Eastern Arabic-Indic (Persian and Urdu)	۰	۱	۲	۳	۴	۵	۶	۷	۸	۹
Devanagari (Hindi)	०	१	२	३	४	५	६	७	८	९
Tamil		௦	௧	௨	௩	௪	௫	௬	௭	௮



Številski sistemi

Dvojiški (binarni) in šestnajstiški (heksadecimalni) sistem

$$(B65F)_{16}$$

$$= 11 * 16^3 + 6 * 16^2 + 5 * 16^1 + 15 * 16^0 =$$

$$= (46687)_{10}$$

$$(1010.011)_2$$

$$= 2^3 + 2^1 + 2^{-2} + 2^{-3} = (10.375)_{10}$$



Številski sistemi

Dvojiški sistem – pretvorba iz desetiškega

$$(41)_{10}$$

$$= 2^5 + 9 = 2^5 + 2^3 + 2^0 = (101001)_2$$

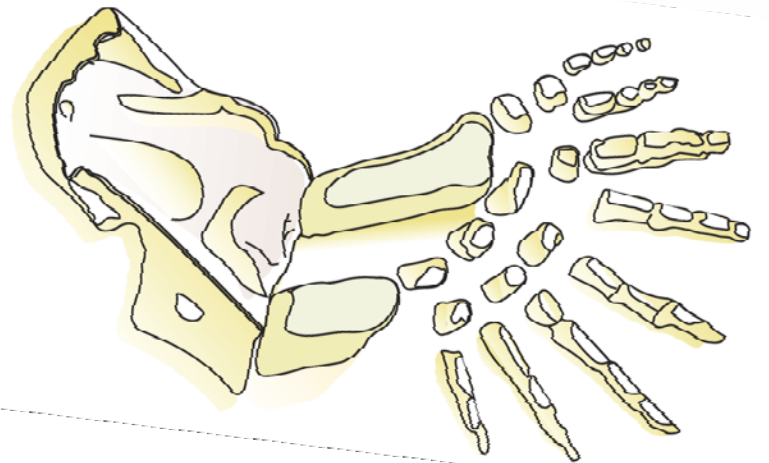


Številski sistemi

Šestnajstiški sistem v (pra)naravi



Acanthostega gunnari, ena prvih štirinožnih živali, je žvela pred pred ca. 365 milijoni let v plitvinah Grenlandije, na sprednjih okončinah pa je imela po osem prstov





Kode in kodiranje

Osnovni pojmi

(digitalno) kodiranje: prirejanje števil, črk in drugih znakov v obliko, ki je primerna za zapis v dvojiškem sistemu

dekodiranje: obratna pretvorba iz kodirane v osnovno obliko

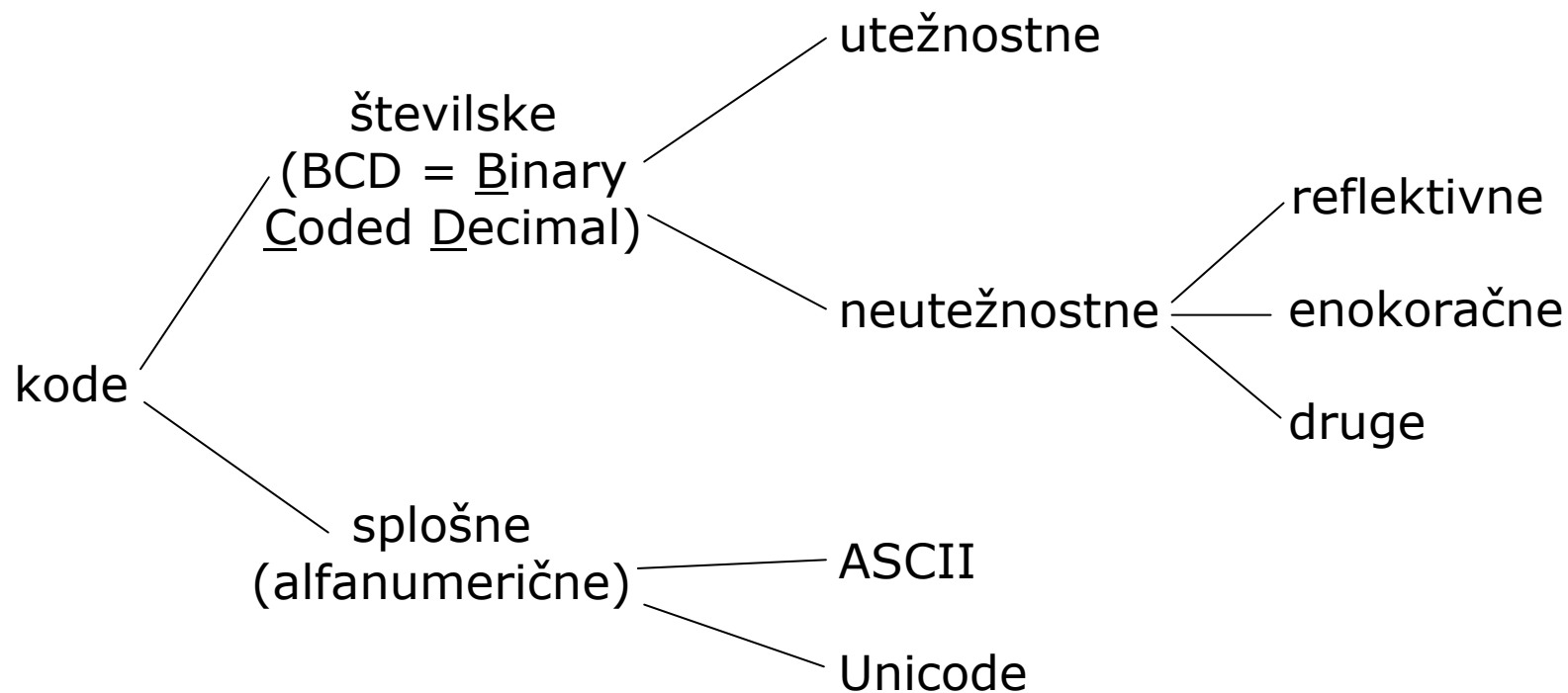
bit: osnovna enota informacije v dvojiškem sistemu

bajt, byte, beseda: skupina osmih bitov



Kode in kodiranje

Delitev kod





Kode in kodiranje

Utežnostne številske kode

število (digit)	naravna BCD (8 4 2 1)	bikvinarna BCD (5 0 4 3 2 1 0)
0	0 0 0 0	0 1 0 0 0 0 1
1	0 0 0 1	0 1 0 0 0 1 0
2	0 0 1 0	0 1 0 0 1 0 0
3	0 0 1 1	0 1 0 1 0 0 0
4	0 1 0 0	0 1 1 0 0 0 0
5	0 1 0 1	1 0 0 0 0 0 1
6	0 1 1 0	1 0 0 0 0 1 0
7	0 1 1 1	1 0 0 0 1 0 0
8	1 0 0 0	1 0 0 1 0 0 0
9	1 0 0 1	1 0 1 0 0 0 0

naravna BCD koda:

- preprosto kodiranje in dekodiranje, tudi "na pamet"
- seštevanje ne deluje vedno:

$$\begin{array}{r} 28 \quad 0010 \quad 1000 \\ +31 \quad +0011 \quad 0001 \\ \hline 59 \quad 0101 \quad 1001 \quad \checkmark \end{array}$$
$$\begin{array}{r} 28 \quad 0010 \quad 1000 \\ +32 \quad +0011 \quad 0010 \\ \hline 60 \quad 0101 \quad 1010 \quad \times \end{array}$$

- koda primerna za prenos številskih podatkov, a sama ne omogoča odkrivanja napak



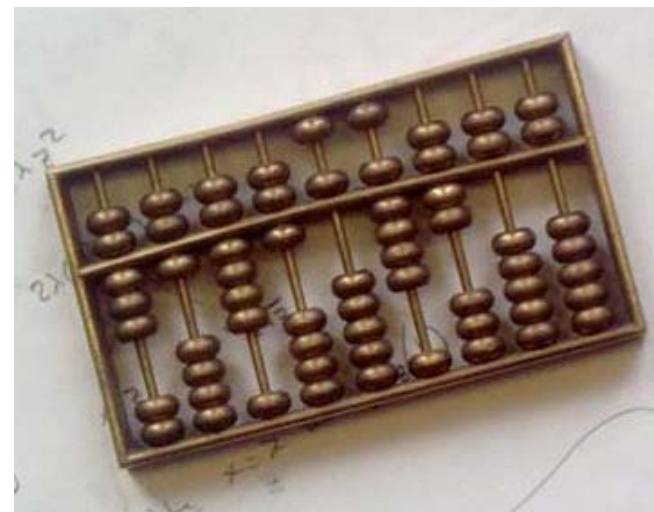
Kode in kodiranje

Utežnostne številske kode

število (digit)	naravna BCD (8 4 2 1)	bikvinarna BCD (5 0 4 3 2 1 0)
0	0 0 0 0	0 1 0 0 0 0 1
1	0 0 0 1	0 1 0 0 0 1 0
2	0 0 1 0	0 1 0 0 1 0 0
3	0 0 1 1	0 1 0 1 0 0 0
4	0 1 0 0	0 1 1 0 0 0 0
5	0 1 0 1	1 0 0 0 0 0 1
6	0 1 1 0	1 0 0 0 0 1 0
7	0 1 1 1	1 0 0 0 1 0 0
8	1 0 0 0	1 0 0 1 0 0 0
9	1 0 0 1	1 0 1 0 0 0 0

bikvinarna BCD koda:

- zaradi lastnosti pomikanja enic je uporabna za štetje ...



abakus Suan-pan, okoli leta 1200



Kode in kodiranje

Utežnostne številske kode

število (digit)	naravna BCD (8 4 2 1)	bikvinarna BCD (5 0 4 3 2 1 0)
0	0 0 0 0	0 1 0 0 0 0 1
1	0 0 0 1	0 1 0 0 0 1 0
2	0 0 1 0	0 1 0 0 1 0 0
3	0 0 1 1	0 1 0 1 0 0 0
4	0 1 0 0	0 1 1 0 0 0 0
5	0 1 0 1	1 0 0 0 0 0 1
6	0 1 1 0	1 0 0 0 0 1 0
7	0 1 1 1	1 0 0 0 1 0 0
8	1 0 0 0	1 0 0 1 0 0 0
9	1 0 0 1	1 0 1 0 0 0 0

bikvinarna BCD koda:

- ... z dodatnimi operacijami pa tudi za aritmetiko



računalnik IBM 650, 1953



Kode in kodiranje

Utežnostne številske kode

število (digit)	naravna BCD (8 4 2 1)	bikvinarna BCD (5 0 4 3 2 1 0)
0	0 0 0 0	0 1 0 0 0 0 1
1	0 0 0 1	0 1 0 0 0 1 0
2	0 0 1 0	0 1 0 0 1 0 0
3	0 0 1 1	0 1 0 1 0 0 0
4	0 1 0 0	0 1 1 0 0 0 0
5	0 1 0 1	1 0 0 0 0 0 1
6	0 1 1 0	1 0 0 0 0 1 0
7	0 1 1 1	1 0 0 0 1 0 0
8	1 0 0 0	1 0 0 1 0 0 0
9	1 0 0 1	1 0 1 0 0 0 0

bikvinarna BCD koda:

- parnost enic omogoča odkrivanje napak brez dodatnih sredstev



Kode in kodiranje

Neutežnostne številske kode

število (digit)	reflektivna BCD koda
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	1 1 0 0
6	1 0 1 1
7	1 0 1 0
8	1 0 0 1
9	1 0 0 0

reflektivna BCD koda:

- zaradi zrcalnosti omogoča preprosto določanje komplementa

$$25 = 0010 \ 1100$$

↕ ↕

$$74 = 1010 \ 0100$$



Kode in kodiranje

Neutežnostne številske kode

število (digit)	enokoračna Grayeva koda
0	0 0 0 0
1	0 0 0 1
2	0 0 1 1
3	0 0 1 0
4	0 1 1 0
5	0 1 1 1
6	0 1 0 1
7	0 1 0 0
8	1 1 0 0
9	1 1 0 1

število (digit)	enokoračna Grayeva koda
10	1 1 1 1
11	1 1 1 0
12	1 0 1 0
13	1 0 1 1
14	1 0 0 1
15	1 0 0 0

Grayeva koda:

- enokoračna (sosednji števili se razlikujeta le enem od štirih bitov)
- reflektivna (če z njo kodiramo šestnajstiška ...
... osmiška ...
... ali štiriška števila)

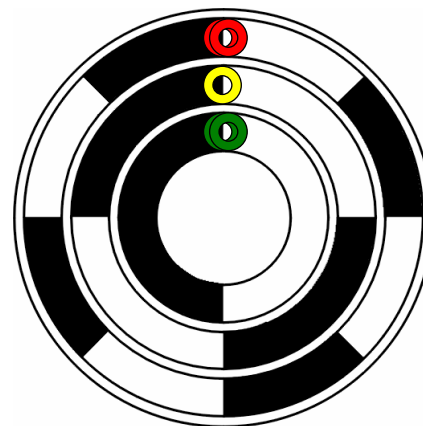


Kode in kodiranje

Kodiranje kota / zasuka

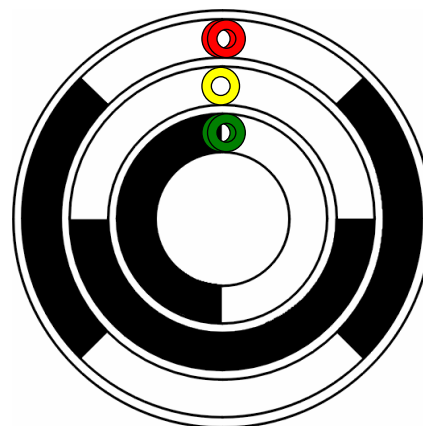
število (digit)	binarna koda	Grayeva koda
0	0 0 0	0 0 0
1	0 0 1	0 0 1
2	0 1 0	0 1 1
3	0 1 1	0 1 0
4	1 0 0	1 1 0
5	1 0 1	1 1 1
6	1 1 0	1 0 1
7	1 1 1	1 0 0

smer vrtenja
štetje navzgor ← → štetje navzdol



binarni kodirnik

0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	1
0	1	0	2	0	1	0	2
0	1	1	3	0	1	1	3
1	0	0	4	1	1	0	6
1	0	1	5	1	0	0	4
1	1	0	6	1	0	0	4
1	1	1	7	1	1	1	7
0	0	0	0	0	1	0	2
0	0	0	0	0	0	0	0



Grayev kodirnik

0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	1
0	1	1	2	0	1	1	2
0	1	0	3	0	1	0	3
1	1	0	4	1	1	0	4
1	1	1	5	1	1	1	5
1	0	1	6	1	0	1	6
1	0	0	7	1	0	0	7
0	0	0	0	0	0	0	0



Kode in kodiranje

Alfanumerične kode – 7-bitna ASCII koda

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	\$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	VEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(56	8	72	H	88	X	104	h	120	x
9	TAB	25	EM	41)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VF	27	ESC	43	+	59	;	75	K	91	[107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	
13	CR	29	GS	45	-	61	=	77	M	93]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL



Kode in kodiranje

Alfanumerične kode – 7-bitna ASCII koda

osmi bit - kontrola parnosti

PAR	64	32	16	8	4	2	1	
0	0	1	1	1	0	0	1	= 57 = "9"
1	1	0	0	0	0	0	0	= 64 = "@"
1	1	0	1	0	1	1	1	= 87 = "W"
0	1	1	1	1	0	1	1	= 123 = "{"
1	1	1	1	1	0	1	1	= NAPAKA !



Kode in kodiranje

Alfanumerične kode – Unicode (UTF-8, UTF-16)

- 8-bitni Unicode: kot ASCII, a brez kontrole paritete znotraj same kode; tako omogoča $2^8 = 256$ različnih znakov namesto $2^7 = 128$, vendar je napake potrebno iskati z dodatnimi kontrolnimi kodami
- 16-bitni Unicode: dve besedi (bajta) namesto ene, s tem omogoča $2^{16} = 65536$ različnih znakov, kar pokrije:
 - latinske črke z diakritiki: à, á, â, ã, ä, å, æ, ã, ā, ...
 - grške črke: α, β, γ, δ, ε, ζ, η, θ, ...
 - cirilske črke: б, в, г, д, ж, з, и, й, л, ...
 - hebrejske črke: א, ב, ט, ה, ז, ...
 - arabske črke in številke: و, ئج, ٥, ٣, ...
 - matematične in druge posebne znake: √, ∞, ∫, ≈, ∂, ☺, ☀, ♀, ♦, ♪, ...



Kode in kodiranje

Odkrivanje in odpravljanje napak

odkrivanje (detekcija): ugotavljamo le, ali je prišlo do napake

odpravljanje (korekcija): določimo, kje je prišlo do napake in napačne podatke nadomestimo s pravilnimi



Kode in kodiranje

Odkrivanje in odpravljanje napak

- z bitom za kontrolo parnosti odkrijemo vsako enojno napako (napako enega bita) v besedi (bajtu)
- če je verjetnost enojne napake majhna, je verjetnost dvojne napake (torej dveh enojnih napak hkrati) še precej manjša:

p - verjetnost, da je v posameznem bitu prišlo do napake,

P_m - verjetnost, da je v posamezni besedi prišlo do m napak,

$$P_m = p^m (1 - p)^{8-m} C_m^8 = p^m (1 - p)^{8-m} \frac{8!}{(8 - m)! m!}$$

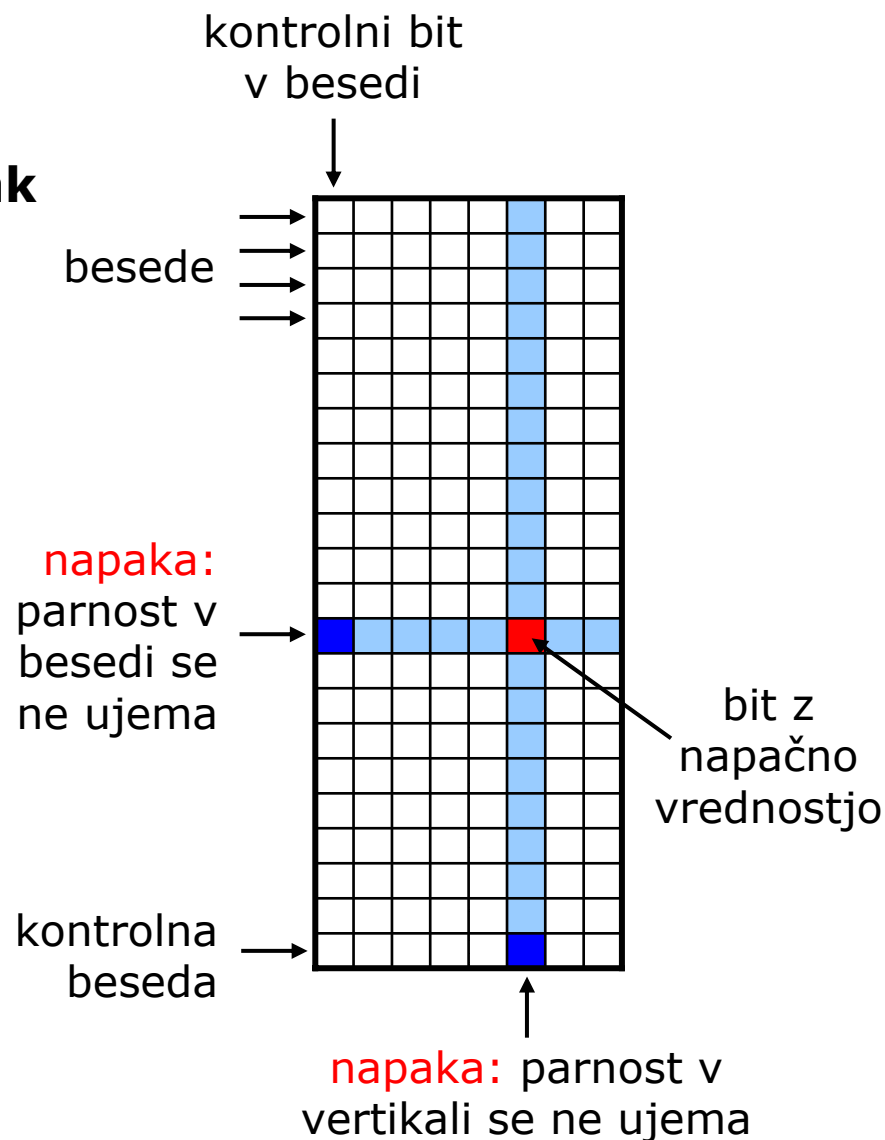
primer: $p = 10^{-4} \Rightarrow P_1 = 7.994 \cdot 10^{-4}, P_2 = 2.798 \cdot 10^{-7}$



Kode in kodiranje

Odkrivanje in odpravljanje napak

- za določitev besede, ki vsebuje napako, kontroli parnosti bitov v besedi ("horizontalni" kontroli) dodamo še "vertikalno" kontrolo parnosti po besedah





Kode in kodiranje

Odkrivanje in odpravljanje napak

- če želimo še večjo natančnost, uvedemo dodatne kontrole (parnost po diagonali, dodatne kontrolne besede, ...)
- v "vertikalni kontroli" se namesto same parnosti uporabljajo tudi vsote po različnih modulih ali komplement takšne vsote (checksum, ciklična kontrola, ...)
- pri večji, predvsem pa spremenljivi pogostosti napak je smiselno prilagajati tudi pogostost kontrole
- prilagodljiva kontrola parnosti je osnova odkrivanja in odpravljanja napak tudi v najučinkovitejših sodobnih kodah za prenos podatkov: turbo kode (mobilna telefonija, satelitske komunikacije), LDPC kode (WiMAX, digitalna televizija)