Kódování pro kanál bez šumu

Příklad 28

Existuje prefixové kódování dekadických číslic 0, 1, 2, …, 9 do abecedy {0,1} takové, že číslice 9 je kódována znakem 0 a ostatní číslice mají kódové značky délky nejvýše pět znaků?

Příklad 28

Takový kód existuje pouze pokud je splněna Kraftova nerovnost:

$$n^{-d_1} + n^{-d_2} + \cdots + n^{-d_{|K|}} \leq 1$$

n - počet znaků kódové abecedy

d_i - délky jednotlivých kódových značek

|K| - počet znaků zdrojové abecedy

Příklad 28

$$n^{-d_1} + n^{-d_2} + \cdots + n^{-d_{|K|}} \leq 1$$

Dosazením:

$$2^{-1} + 9 \cdot 2^{-5} = \frac{1}{2} + \frac{9}{32} = \frac{25}{32} \leq \mathbf{1}$$

Závěr:

Takový kód existuje.

Příklad 28

Pokusme se kód sestrojit:

9	0	0
8	1xxxx	1000
7	1xxxx	1001
6	1xxxx	1010
5	1xxxx	1011
4	1xxxx	1100
3	1xxxx	1101
2	1xxxx	1110
1	1xxxx	11110
0	1xxxx	11111

Je zřejmé, že čtyři binární pozice na rozlišení devíti znaků postačí.

Tři binární pozice by stačily na rozlišení osmi znaků.

Osmý a devátý znak rozlišíme další binární cifrou.

Příklad 28

Dva z možných kódů vyhovujících zadání:

9	0	0
8	10000	1000
7	10001	1001
6	10010	1010
5	10011	1011
4	10100	1100
3	10101	1101
2	10110	1110
1	10111	11110
0	11000	11111

Příklad 29

Vytvořte tříznakové prefixové kódy, které bude kódovat abecedu {A, B, C, D, E, F} do abecedy {0, 1, 2}.

Kód K_1 : Dvě písmena budou kódována značkou délky 1, čtyři písmena značkou délky 2.

Kód K_2 : Dvě písmena budou kódována značkou délky 1, dvě písmena značkou délky 2 a dvě písmena značkou délky 3.

Příklad 29

Kód K_1 :

Ověření existence kódu Kraftovou nerovností:

$$n^{-d_1} + n^{-d_2} + \cdots + n^{-d_{|K|}} \leq 1$$

Dosazením:

$$2 \cdot 3^{-1} + 4 \cdot 3^{-2} = \frac{2}{3} + \frac{4}{9} = \frac{6+4}{9} \geq 1$$

Závěr:

Takový kód neexistuje, nelze vytvořit.

Příklad 29

Kód K_2 :

Ověření existence kódu Kraftovou nerovností:

$$n^{-d_1} + n^{-d_2} + \cdots + n^{-d_{|K|}} \leq 1$$

Dosazením:

$$2 \cdot 3^{-1} + 2 \cdot 3^{-2} + 2 \cdot 3^{-3} = \frac{2}{3} + \frac{2}{9} + \frac{2}{27} =$$

$$= \frac{18 + 6 + 2}{27} = \frac{26}{27} \leq \mathbf{1}$$

Příklad 29

Kód K_2 :

Závěr:

Kód s délkami kódových značek 1,1,2,2,3,3 existuje, lze jej sestrojit.

Příklad 29

Sestrojení kódu K_2 :

Abeceda zdroje {A, B, C, D, E, F}

Kódová abeceda {0, 1, 2}

Délky kódových značek 1,1,2,2,3,3

A	0
B	1
C	20
D	21
E	220
F	221

Příklad 29

Vytvořte KA pro dekódování kódu K_2 :

Množina stavů automatu bude odpovídat prefixům kódových značek s výjimkou kódových značek samotných.

Příklad 29

Vytvořte KA pro dekódování kódu K_2 :

		prefixy kódových značek
A	0	e, 0
B	1	e, 1
C	20	e, 2, 20
D	21	e, 2, 21
E	220	e, 2, 22, 220
F	221	e, 2, 22, 221

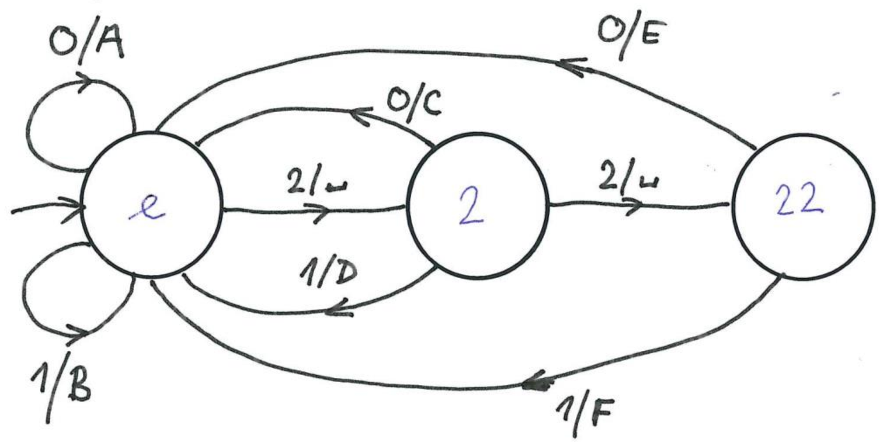
KA bude mít tři stavy, budou odpovídat prefixům

e, 2, 22.

Příklad 29

Konečný automat pro dekódování

A	0
B	1
C	20
D	21
E	220
F	221



Příklad 29

Přidejme do zdrojové abecedy K_2 písmeno G:

A	0
B	1
C	20
D	21
E	220
F	221
G	222

a ověřme Kraftovu nerovnost.

Příklad 29

A	0
B	1
C	20
D	21
E	220
F	221
G	222

Pravá strana Kraftovy nerovnosti je rovna 1, což souvisí s tím, že jsou využity všechny potenciálně možné kódové značky.

$$2 \cdot 3^{-1} + 2 \cdot 3^{-2} + 3 \cdot 3^{-3} = \frac{2}{3} + \frac{2}{9} + \frac{3}{27} =$$

$$= \frac{18 + 6 + 3}{27} = \frac{27}{27} = 1$$

Huffmanova konstrukce kódu s minimální střední délkou značky

Příklad 30

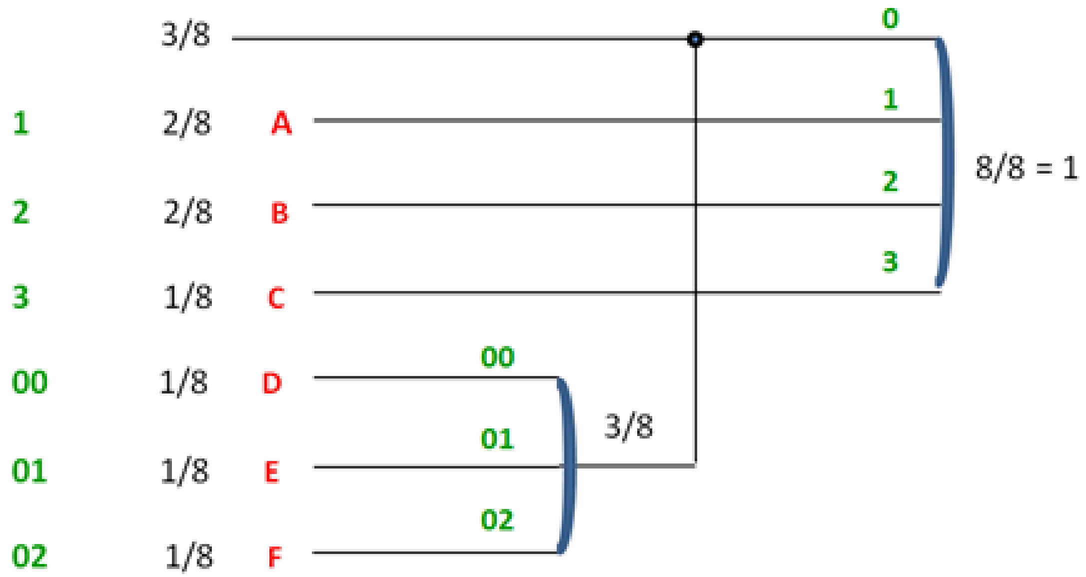
Je dána abeceda {A, B, C, D, E, F} s pravděpodob-nostmi výskytu znaků {2/8, 2/8, 1/8, 1/8, 1/8, 1/8}.

Kódová abeceda je {0, 1, 2, 3}.

Najděte kód s minimální střední délkou kódové značky.

Příklad 30

5. Výsledné přiřazení kódových značek:



Příklad 31

Je dána abeceda a pravděpodobnosti výskytu jednotlivých znaků v procentech:

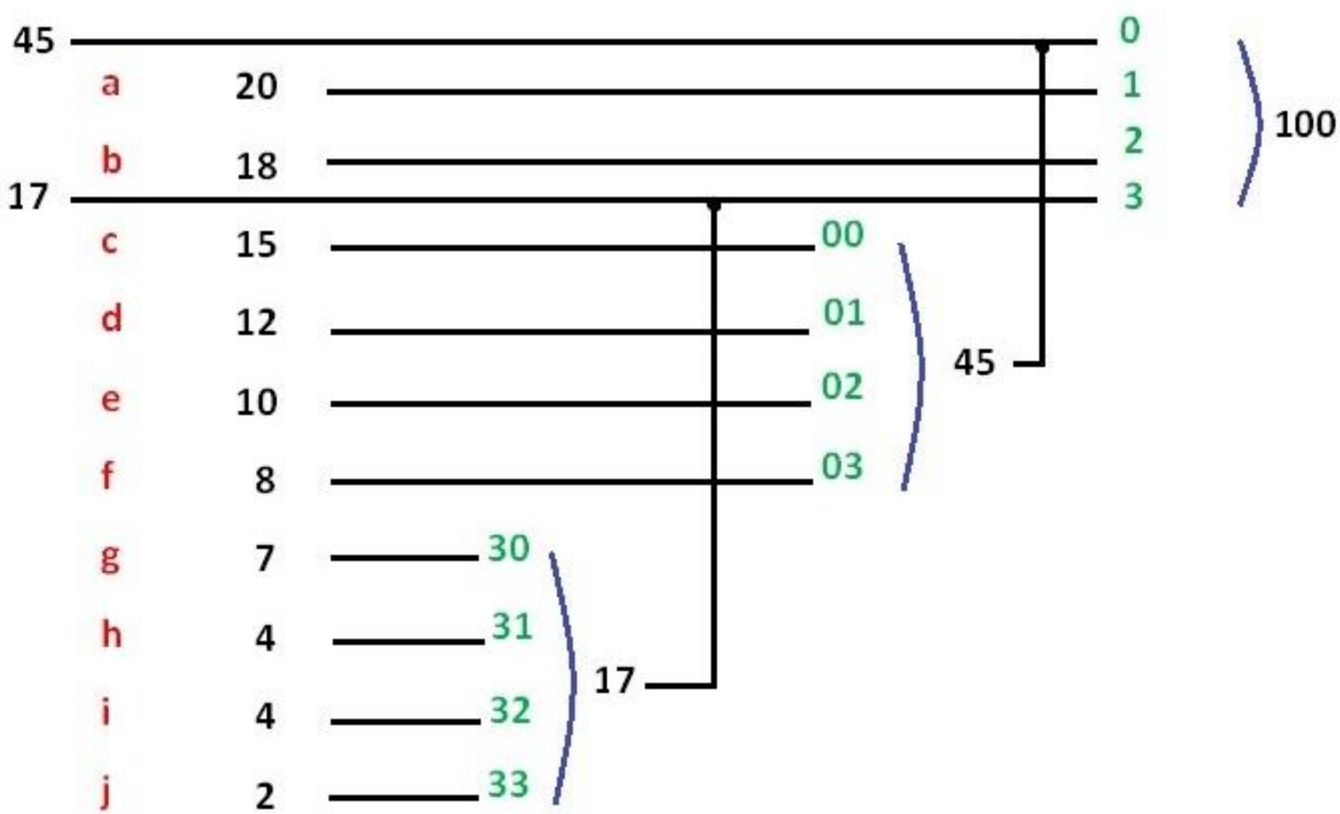
$A = \{a, \quad b, \; c, \; d, \; e, f, \; g, h, \; i, \; j\}$

$p(A) = (20, 18, 15, 12, 10, 8, 7, 4, 4, 2)$

Kódová abeceda je $\{0, 1, 2, 3\}$.

Najděte kód s minimální střední délkou značky.

Příklad 31



45
a
20
0
1
b
18
2
100
17
3
c
15
00
d
12
01
45
e
10
02
f
8
03
g
7
30
h
4
31
17
i
4
32
j
2
33

Příklad 31

Kód s minimální střední délkou kódové značky:

a	**1**
b	**2**
c	**00**
d	**01**
e	**02**
f	**03**
g	**30**
h	**31**
i	**32**
j	**33**

Příklad 31

Výpočet středního počtu jednotlivých znaků ve značce:

			0	**1**	**2**	**3**	
a	**1**	0.20	0	0.2	0	0	
b	**2**	0.18	0	0	0.18	0	
c	**00**	0.15	0.30	0	0	0	
d	**01**	0.12	0.12	0.12	0	0	
e	**02**	0.10	0.10	0	0.10	0	
f	**03**	0.08	0.08	0	0	0.08	
g	**30**	0.07	0.07	0	0	0.07	
h	**31**	0.04	0	0.04	0	0.04	
i	**32**	0.04	0	0	0.04	0.04	
j	**33**	0.02	0	0	0	0.04	
			0.67	**0.36**	**0.32**	**0.27**	**1,62**

Příklad 31

Výpočet redundance ve zdrojové i kódové abecedě:

	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
a	0,20		-2,321928	-0,464386
b	0,18		-2,473931	-0,445308
c	0,15		-2,736966	-0,410545
d	0,12		-3,058894	-0,367067
e	0,10		-3,321928	-0,332193
f	0,08		-3,643856	-0,291508
g	0,07		-3,836501	-0,268555
h	0,04		-4,643856	-0,185754
i	0,04		-4,643856	-0,185754
j	0,02		-5,643856	-0,112877
	1,00		**H(X)**	**3,063947**
			ρ	**7,77%**

	N_i	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
0	0,67	0,41		-1,273761	-0,526802
1	0,36	0,22		-2,169925	-0,482206
2	0,32	0,20		-2,339850	-0,462193
3	0,27	0,17		-2,584963	-0,430827
	1,62	1,00		**H(X)**	**1,902028**
				ρ	**4,90%**

Příklad 31

Výpočet redundance ve zdrojové i kódové abecedě:

	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
a	0,20		-2,321928	-0,464386
b	0,18		-2,473931	-0,445308
c	0,15		-2,736966	-0,410545
d	0,12		-3,058894	-0,367067
e	0,10		-3,321928	-0,332193
f	0,08		-3,643856	-0,291508
g	0,07		-3,836501	-0,268555
h	0,04		-4,643856	-0,185754
i	0,04		-4,643856	-0,185754
j	0,02		-5,643856	-0,112877
	1,00		**H(X)**	**3,063947**
			ρ	**7,77%**

	N_i	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
0	0,67	0,41		-1,273761	-0,526802
1	0,36	0,22		-2,169925	-0,482206
2	0,32	0,20		-2,339850	-0,462193
3	0,27	0,17		-2,584963	-0,430827
	1,62	1,00		**H(X)**	**1,902028**
				ρ	**4,90%**

Nešlo by udělat něco, aby bylo rozložení p-stí znaků 0,1,2,3 rovnoměrnější?

Příklad 31

Výpočet redundance ve zdrojové i kódové abecedě:

	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
a	0,20		-2,321928	-0,464386
b	0,18		-2,473931	-0,445308
c	0,15		-2,736966	-0,410545
d	0,12		-3,058894	-0,367067
e	0,10		-3,321928	-0,332193
f	0,08		-3,643856	-0,291508
g	0,07		-3,836501	-0,268555
h	0,04		-4,643856	-0,185754
i	0,04		-4,643856	-0,185754
j	0,02		-5,643856	-0,112877
	1,00		**H(X)**	**3,063947**
			ρ	**7,77%**

	N_i	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
0	0,67	0,41		-1,273761	-0,526802
1	0,36	0,22		-2,169925	-0,482206
2	0,32	0,20		-2,339850	-0,462193
3	0,27	0,17		-2,584963	-0,430827
	1,62	1,00		**H(X)**	**1,902028**
				ρ	**4,90%**

Nešlo by udělat něco, aby bylo rozložení p-stí znaků 0,1,2,3 rovnoměrnější?

Ano. Zaměnit přiřazení kódových značek se stejnými délkami.

Příklad 31

Ekvivalentní kód:

			0	1	2	3	
a	1	0.20	0	0.2	0	0	
b	2	0.18	0	0	0.18	0	
c	03	0.15	0.15	0	0	0.15	
d	01	0.12	0.12	0.12	0	0	
e	02	0.10	0.10	0	0.10	0	
f	00	0.08	0.16	0	0	0	
g	33	0.07	0	0	0	0.14	
h	31	0.04	0	0.04	0	0.04	
i	32	0.04	0	0	0.04	0.04	
j	30	0.02	0.02	0	0	0.02	
			0.55	**0.36**	**0.32**	**0.39**	1,62

Příklad 31

Výpočet redundance ve zdrojové i kódové abecedě:

	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
a	0,20		-2,321928	-0,464386
b	0,18		-2,473931	-0,445308
c	0,15		-2,736966	-0,410545
d	0,12		-3,058894	-0,367067
e	0,10		-3,321928	-0,332193
f	0,08		-3,643856	-0,291508
g	0,07		-3,836501	-0,268555
h	0,04		-4,643856	-0,185754
i	0,04		-4,643856	-0,185754
j	0,02		-5,643856	-0,112877
	1,00		**H(X)**	**3,063947**
			ρ	**7,77%**

	N_i	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
0	0,67	0,41		-1,273761	-0,526802
1	0,36	0,22		-2,169925	-0,482206
2	0,32	0,20		-2,339850	-0,462193
3	0,27	0,17		-2,584963	-0,430827
	1,62	1,00		**H(X)**	**1,902028**
				ρ	**4,90%**

	N_i	p_i		$\log_2(p_i)$	$p_i \cdot \log_2(p_i)$
0	0,55	0,34		-1,558490	-0,529117
1	0,36	0,22		-2,169925	-0,482206
2	0,32	0,20		-2,339850	-0,462193
3	0,39	0,24		-2,054448	-0,494589
	1,62	1,00		**H(X)**	**1,968105**
				ρ	**1,59%**

Příklad 31

Závěr:

Huffmannův algoritmus nalezne kód s minimální střední délkou kódové značky.

Tento kód ale nemusí mít minimální redundanci.

Ekvivalentní kód s minimální redundancí lze nalézt „vhodnou záměnou“ kódových značek se stejnou délkou.