NTIN 100 Základy přenosu a zpracování informace **léto 2017/2018**

3. domácí úlohy - samoopravné kódy

do 7. května 2018

Úloha 1. Nechť G_1 a G_2 jsou generující matice kódů s parametry $[n_1, k, d_1]_q$ a $[n_2, k, d_2]_q$. Určete a zdůvodněte, jaké kódy generují následující matice

a)

$$\begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix}$$

b)

$$\begin{pmatrix} G_1 & G_2 \end{pmatrix}$$

c)

$$G_1 \otimes G_2 = \begin{pmatrix} a_{1,1}G_2 & a_{1,2}G_2 & \cdots & a_{1,n_1}G_2 \\ a_{2,1}G_2 & a_{2,2}G_2 & \cdots & a_{2,n_1}G_2 \\ \cdots & \cdots & \cdots & \cdots \\ a_{k,1}G_2 & a_{k,2}G_2 & \cdots & a_{k,n_1}G_2 \end{pmatrix}.$$

Zde

$$G_1 = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n_1} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n_1} \\ \cdots & \cdots & \cdots & \cdots \\ a_{k,1} & a_{k,2} & \cdots & a_{k,n_1} \end{pmatrix}$$

a $a_{i.j}G_2$ je matice G_2 vynásobená po složkách skalárem $a_{i,j}$.

Úloha 2. Pro dané n a $d = \delta n$, sestrojme (n, k, d)-kód $C \subseteq \{0, 1\}^n$ hladově, tedy tak, že do původně prázdného C přidáváme další kódová slova tak dlouho, dokud to lze, aniž bychom porušili minimální vzdálenost d. Jak velké bude k alespoň? Kolik chyb bude C opravovat? Srovnejte to s kódem ze Shannonovy věty.

Úloha 3. Ukažte, že pro každý (n, k, d)-kód platí: $d \leq n - k + 1$.

Úloha 4. V Reed-Solomonově kódu se zpráva $m = m_1 m_2 \cdots m_k \in GF[q]$ interpretuje jako koeficienty polynomu $p_m(x)$ a kódem pro m je $(p_m(\alpha_1), \ldots, p_m(\alpha_n))$. Ukažte, že pokud m přiřadíme polynom $p'_m(x)$ stupně nejvýše $k-1$ takový, že $p'_m(\alpha_i) = m_i$, pro $i = 1, \ldots, k$, a $(p'_m(\alpha_1), p'_m(\alpha_2), \ldots, p'_m(\alpha_n))$ prohlásíme za kód m, pak dostaneme opět Reed-Solomonův kód. Nalezněte generující matici takového kódu.

Úloha 5. Nechť n je kladné celé číslo. Zkonstruujme následující kód: nechť zpráva M je matice z $GF[2]^{n \times n}$. Její zakódování je M společně s paritou každého řádku, paritou každého sloupce a paritou těchto parit (tedy matice (vektor) z $GF[2]^{(n+1)^2}$). Kolik chyb tento kód umí opravit? Jak chyby opravovat?