závěrečná písemka 5.1.2022, skupina **A**

Obecné doporučení: Čtěte pozorně, často jsou v textu informace, které Vám počítání zjednoduší. Navíc se po Vás také často chce několik věcí, tak na žádnou nezapomeňte.

1. Lukáš se zabýval počítáním odmocnin modulo 49.

a) Z učebnice vyčetl, že primitivní kořen modulo 49 existuje a že k jeho nalezení se mu bude hodit primitivní kořen modulo 7, kterým je např. 3. **Poraďte mu**, jakou (jednu!) mocninu čísla 3 modulo 49 má spočítat (sami ji nepočítejte!), aby zjistil, zda je 3 primitivní kořen dokonce modulo 49. Možná byste chtěli spočítat tři mocniny, ale dvě z nich stačí uvážit modulo 7.

b) Prvně zkusil využít Jacobiho symbolu $\left(\frac{x}{49}\right)$ k ověření, zda dané x, nesoudělné s 49, je druhou mocninou. **Vysvětlete**, proč je to zcela nevhodné. **Najděte** nějaké takové x, pro které sice $\left(\frac{x}{49}\right) = 1$, ale druhou mocninou není (můžete také využít dalšího bodu).

c) Dále se dočetl, že odmocniny lze počítat snadno pomocí diskrétního logaritmu $\log_3$ vzhledem k primitvnímu kořenu 3. **Poraďte mu**, jak pomocí $\log_3 x$ zjistit, zda x je druhou mocninou modulo 49 a jak spočítat druhé odmocniny x (stačí jako vhodné mocniny primitivního kořene 3). **Demonstrujte** na příkladu $x \equiv 30 \pmod{49}$ s vypočteným $\log_3 30 = 14$ (odmocniny není potřeba dopočítávat explicitně).

(12 bodů)

2. V šifrovacím systému RSA s veřejným klíčem daným modulem $n = 247 = 13 \cdot 19$ a exponentem $e = 29$ je zašifrována zpráva. **Dešifrujte** zprávu $c = 35$, přičemž počítejte **prvně zvlášť** modulo 13 a modulo 19 a tyto výpočty pak **dejte dohromady** (pomocí CRT). **(12 bodů)**

3. Polynomem $1 + x + x^3$ lze generovat lineární $(k+3, k)$-kód pro libovolné k. **Pro které** nejmenší k se vrchní tři prvky prvního sloupce matice kódu, tj. 110, poprvé zopakují v nějakém dalším sloupci? (Tj. generujte matici, dokud nedostanete znovu sloupec začínající 110.) Uvažujme dále toto k, tj. šířku matice.

a) **Vysvětlete**, na příkladu obdrženého slova $110 \mid 0 \cdots 0$, že výsledný kód nezvládne opravovat jednoduché chyby.

b) V případě obdrženého slova $010 \mid 1010 \cdots 0$ však v něm obsaženou jednouduchou chybu opravit zvládne, **ukažte** jak. (klasické dekódování)

c) Určené k není nejmenší, při kterém výsledný kód nezvládne opravovat jednoduché chyby. **Jaká je** tato nejmenší šířka matice?

(12 bodů)

4. Cílem tohoto příkladu bude spočítat průměrný počet hodů jednou šestistěnnou kostkou potřebných na hození dvou šestek po sobě.

a) Označme a_k pravděpodobnost, že v k po sobě jdoucích hodech nepadne šestka dvakrát po sobě a navíc poslední hod *je* šestka. Podobně označme b_k pravděpodobnost, že v k po sobě jdoucích hodech nepadne šestka dvakrát po sobě a navíc poslední hod *není* šestka. **Sestavte** soustavu rekurencí pro tyto posloupnosti, počáteční podmínky budou $a_0 = 0$, $b_0 = 1$. **Najděte** vytvořující funkce $A(x)$, $B(x)$ těchto posloupností, o explicitní vzoreček pro a_k, b_k se nesnažte (nebude hezký a nebudete ho potřebovat).

b) Označme dále p_k pravděpodobnost, že v k po sobě jdoucích hodech padne šestka dvakrát po sobě, ale až v úplně posledních dvou hodech a označme $P(x)$ vytvořující funkci této posloupnosti. **Dokažte**, že derivace funkce $P(x)$ v čísle 1 je rovna

$$P'(1) = \sum_{k \geq 0} p_k \cdot k$$

a je tedy rovna průměrnému počtu hodů jednou šestistěnnou kostkou potřebných na hození dvou šestek po sobě. Zjevně $p_k = \frac{1}{6} \cdot a_{k-1}$ a vytvořující funkce proto bude $P(x) = \frac{1}{6} \cdot x \cdot A(x)$.

c) **Spočtěte** tuto průměrnou hodnotu (buď jako derivaci z předchozího bodu nebo jinak, ale úplně zadarmo to asi nepůjde).

(18 bodů)