

Věta 1. Bud'te X_1, \dots, X_n nezávislé binární náhodné proměnné s $\Pr[X_i = 1] = p_i$. Označme $X = \sum_{i=1}^n X_i$ a $\mu = \mathbb{E}[X] = \sum_{i=1}^n p_i$. Potom platí, že

- $\Pr[X \leq (1 - \delta)\mu] \leq e^{-\mu\delta^2/2}$ pro všechna $0 < \delta < 1$ a
- $\Pr[X \geq (1 + \delta)\mu] \leq e^{-\frac{\delta^2\mu}{2+\delta}}$ pro všechna $\delta > 0$.

Příklady

- Mějme dvě mince. Na první padne hlava s pravděpodobností $p_1 = 1/2$ a na druhé s pravděpodobností $p_2 = 1/4$. Jak určíme, která je která? Použijme následující algoritmus: Hodíme libovolnou mincí n -krát a určíme pravděpodobnost \hat{p} s jakou během našeho algoritmu padala hlava. Pokud $\hat{p} \geq 3/8$ řekneme, že mince se kterou jsme házeli je první mince, jinak řekneme, že je to mince druhá.
 - Ukažte, že pro $n \geq 64 \ln(2/\delta)$ náš algoritmus odpoví správně s pravděpodobností alespoň $1 - \delta$.
 - Co by se stalo, pokud bychom neznali pravděpodobnost p_2 ?
- Ukažte, že pro randomizovaný algoritmus QuickSort a jeho čas běhu platí $\Pr[\text{runtime} > 8n \log n] < 1/n^2$.
- Jak simulovat spravedlivou minci cinklou a naopak? Jaký je očekávaný počet hodů?
 - Mějme spravedlivou minci. Chceme pomocí ní simulovat hod mincí na které padá hlava s pravděpodobností $0 < p < 1$.
 - Mějme minci na které padá hlava s pravděpodobností $0 < p < 1$. Chceme pomocí ní simulovat hod spravedlivou mincí.
- Navrhněte algoritmus, který pro daný bipartitní graf G rozhodne, zda počet perfektních párování grafu G je sudý nebo lichý.
- Bud' $A \in \{0, 1\}^{n \times n}$ matice. Bud'te $\varepsilon_{i,j} \in \{-1, 1\}$ nezávislé náhodné proměnné pro $1 \leq i, j \leq n$. Dále B je náhodná matice, pro kterou platí $b_{i,j} = \varepsilon_{i,j} \cdot a_{i,j}$ (tedy jsme náhodně rozhodili znaménka do matice A).
 - Ukažte, že $\mathbb{E}[\det(B)] = 0$.
 - Ukažte, že $\mathbb{E}[\det(B)^2] = \text{per}(A)$.
- Ukažte, že $\text{per}(A) = \sum_{i=1}^n a_{1,i} \cdot \text{per}(A_{1,i})$, kde A je $n \times n$ matice a $A_{1,i}$ je $(n-1) \times (n-1)$ matice, která vznikne z A smazáním prvního řádku a i -tého sloupce.