Pravidlá

Riešenia tejto písomnej skúšky je potrebné odovzdať do **piatku 6. 1. 2017, 10:00**, a to vo formáte PDF e-mailom na `misof@ksp.sk`. Počas riešenia priebežne **sledujte oznamy** na webstránke predmetu.

Smiete využívať ľubovoľné neživé zdroje informácií ktoré existovali v okamihu začiatku písomky. Pochopiteľne, vrátane všetkého zverejneného na stránke predmetu. Až do deadline (a to bez ohľadu na to, kedy odovzdáte svoje riešenia) je zakázané akýmkoľvek spôsobom diskutovať o úlohách s kýmkoľvek živým.
Použité externé zdroje dostatočne **adekvátne citujte**. Čo viete odcitovať, netreba rozpisovať.
Do hodnotenia sa vám započíta **5 najlepšie vyriešených** úloh. Zápisaniu známky môže predchádzať rozhovor o niektorých úlohách ktoré ste riešili.

0 Nultá podúloha

Do textu riešenia nezabudnite prosím vyplniť dva údaje: celé meno a **odhadovaný čistý čas** (v hodinách) strávený riešením úloh. Tento údaj **nebude mať vplyv na hodnotenie**, chcem ho kvôli lepšej kalibrácii úloh v budúcnosti.

1 Triangel

Daný je jednoduchý neorientovaný graf s n vrcholmi. Graf je hustý, teda má $\Theta(n^2)$ hrán.
Trojuholník je neusporiadaná trojica navzájom rôznych vrcholov $\{u, v, w\}$ taká, že každé dva z nich sú spojené hranou.
Navrhnite ľubovoľný algoritmus, ktorý spočíta trojuholníky v našom grafe v lepšom ako kubickom čase.

2 Goldbach

Overte Goldbachovu domnienku po $n = 10^7$.
Presnejšie, pre každé párne n od 4 po 10^7 zistite hodnotu $s(n)$: počet dvojíc prvočísel (p, q) takých, že $p + q = n$.
Ak $p \neq q$, dvojicu (q, p) považujeme za odlišnú od (p, q). Napr. $s(10) = 3$ lebo $10 = 3 + 7 = 5 + 5 = 7 + 3$.
Pre rýchlejšiu kontrolu na konci celého procesu ešte spočítajte a odovzdajte hodnotu $\sum(s(n))^2$.
Hodnotí sa aj efektívnosť použitého algoritmu. Partial credit je za popis algoritmu bez zistenia požadovanej hodnoty.

3 Mini shuffles

Hovoríme, že permutácia π je k-lokálna, ak platí $\forall i : |\pi(i) - i| \leq k$.
Nech f_k je funkcia, ktorá každému n priradí počet k-lokálnych permutácií množiny $\{0, 1, \dots, n-1\}$.
Zistite asymptotickú rýchlosť rastu funkcií f_1, f_2 a f_3.
(Ak to zistíte len pre niektoré z nich, dostanete značný partial credit. Ak by ste sa naopak nudili, môžete sa ešte nesúťažne zamyslieť nad tým, pre aké k (v závislosti od n) bude už $f_k \sim n!$.)

4 Kamionem po Evropě

Keď sa v Žarnovici pri motoreste pristaví kamión, servírka Zuzanka sa vždy šoféra opýta, odkiaľ a kam ide. Obe miesta potom na mape sveta označí špendlíkom a teší sa z toho, aké exotické končiny už má označené.
Pre jednoduchosť predpokladajme, že kamión idúci z A do B vždy použije najkratšiu cestu. Neformálne si *prestíž* Žarnovice môžeme definovať ako maximálnu možnú cestnú vzdialenosť medzi Žarnovicou a miestom označeným špendlíkom.
Formálne definujte *prestíž* vrcholu v grafe. Ako súvisí prestíž a dosah (reach) vrcholu v grafe? Vysvetlite, či a ako sa dá použiť predpočítané prestíže vrcholov na zrýchlenie algoritmu hľadajúceho najkratšiu cestu z A do B, a zamyslite sa stručne nad tým, či je to dobrý nápad.

5 Pre-NTT

Niekedy v budúcnosti chceme použiť NTT (teda DFT nad konečným poľom $\mathbb{Z}_p$) na napísanie programu, ktorý bude vedieť medzi sebou vynásobiť dva polynómy, ktorých stupeň je do 15 000 000 a ktorých koeficienty sú celé čísla z $[0, 10^{15}]$. Nájdite vhodné parametre pre NTT: nejaké vhodné p, k nemu nejaké vhodné ω a n (kde ω je vhodná n-tá odmocnina z 1 v $\mathbb{Z}_p$). Presvedčte ma o korektnosti a vhodnosti vami zvolených parametrov.

6 Pažravý NIM

Hráme NIM s n kôpkami kamienkov. Na rozdiel od klasického NIMu však používame upravené pravidlá: hráč na ťahu si vždy musí vybrať niektorú neprázdnu kôpku a odobrať z nej *aspoň dve tretiny* kamienkov, ktoré sa na nej v danej chvíli nachádzajú. (Napr. ak si vybral kôpku so 7 kamienkami, môže ich odobrať 5, 6, alebo všetkých 7.) Ako v klasickom NIMe, aj tu platí, že prehráva ten, kto už nevie spraviť platný ťah.
Navrhnite čo najefektívnejší algoritmus, ktorý pre danú pozíciu tejto hry zistí, ktorý z hráčov má vyhrávajúcu stratégiu.

7 Hamilton

Daný je orientovaný graf, v ktorom má každý silno súvislý komponent veľkosť nanajvýš 10. Navrhnite algoritmus, ktorý v polynomiálnom čase zistí, či tento graf obsahuje Hamiltonovskú cestu.
Partial credit / hint: nahraďte si číslo 10 vhodne zvolenou menšou konštantou :)