



P1 Losí utkání (řešení)



InterLoS 2017

V tejto úlohe výnimočne nebolo potrebné dokonalé riešenie. Niekoľko možných riešení, ktoré bolo možné zkúsiť: vyberať losy náhodne, prečítať čísla, ktoré už súper použil a použiť to, čo ma najvyššiu pravdepodobnosť uspieť, simulovať, ako dopadne hra a vybrať podľa výsledku, snažiť sa naučiť súperovú stratégiu, ...

Zápasov sa účastnilo aj zopár našich botov, aby ste mali s kým hrať aj keď práve neboli dostatok súperov. Ale po zvyšok času ste hrali medzi sebou. Preto bolo ťažké odhadnúť aké zložité budú stratégie. Jednu jednoduchú stratégiu nájdete v súbore *P1s-solution.php*.

Výsledné heslo: *MOUKA*



P2 Los s ručením omezeným (řešení)



InterLoS 2017

Úlohu lze vyřešit několika způsoby. Pravděpodobně ten nejméně pracný je si uvědomit, že se jedná o nějaké přesuny jednotek v grafu¹ – o toky.² Úlohu tedy budeme řešit pomocí toků v grafech.

Musíme si však uvědomit, že se nejedná o nalezení libovolného toku, ale toku s minimální cenou.³ Po zadání našeho problému do Googlu zjistíme, že jedním z algoritmů pro řešení toků s minimální cenou je algoritmus s názvem Network simplex.⁴

Jelikož výpočet bude jen přímým spuštěním již existujícího algoritmu na konkrétním grafu, tak nám jen zbývá si zvolit konkrétní nástroj pro implementaci. Pro autorské řešení je zvolen programovací jazyk Python a knihovna NetworkX (<https://networkx.github.io>). S těmito nástroji napíšeme výsledný kód cca na 20 řádků. Vzorové řešení naleznete v souboru *P2s-solution.py*.

Výsledné heslo: *1258*

¹Graf: [https://en.wikipedia.org/wiki/Graph_\(discrete_mathematics\)](https://en.wikipedia.org/wiki/Graph_(discrete_mathematics))

²Tok v grafu: https://en.wikipedia.org/wiki/Flow_network

³Tok s minimální cenou: https://en.wikipedia.org/wiki/Minimum-cost_flow_problem

⁴Network Simplex: https://en.wikipedia.org/wiki/Network_simplex_algorithm



P3 Sekvence (řešení)



InterLoS 2017

V této úloze je třeba napsat simulátor [Turingova stroje](#) a spustit ho s pravidly a vstupem ze zadání. Ač Turingův stroj obecně pracuje s nekonečnou páskou, zde vám vždy stačila páska o délce $2 + \text{délka vstupu}$ (2 pro počáteční a koncovou značku).

V první části stačil simulátor, na němž spuštěný program běží až do okamžiku, kdy neexistuje žádné pravidlo, kterým se dá pokračovat.

V druhé části pak bylo třeba rozšířit simulátor tak, aby si pamatoval, jaké konfigurace, tedy kombinace *obsah pásky + stav + pozice na pásce* již viděl. Výsledkem pak byla první opakující se konfigurace. Ta měla na pásce následující:

```
>posvatne.losi.slovo.ktere.chces.je.vypinac.toto.slovo.pouzij_
```

Vzorové řešení naleznete v souboru *P3s-solution.py*.

Výsledné heslo: *VYPINAC*



P4 Galton (řešení)



InterLoS 2017

Táto úloha má samozrejme niekoľko možných riešení. Možno najjednoduchšie, avšak nepresné, je simulovať nejaký počet guľičiek náhodne padajúcich cez mriežku. Pokiaľ budete simulovať dostatočný počet guľičiek a na každom kolíku, kde sa treba, sa budete rozhodovať náhodne, mali by ste dostať riešenie dostatočne presné na získanie správnej odpovede.

Avšak, táto úloha má aj riešenie exaktné a teda by sme sa mohli pýtať aj na presné vyjadrenie pravdepodobností jednotlivých prepážok. Toto riešenie začína zdola hracej mriežky a využíva fakt, že pravdepodobnosti všetkých prepážok dosiahnuteľných z nejakého kolíka sa dajú vypočítať z vedomosti pravdepodobností prepážok dosiahnuteľných z kolíkov priamo pod ním (vľavo a vpravo). Riešenie tak začína na spodnej strane mriežky.

Predstavme si len jeden kolík a pod ním dve prepážky do ktorých môže guľička spadnúť. Ak kolík nieje na spodnej strane nejakej trojuholníkovej zarážky (ktorá by umožňovala guľičkám padať iba na jednu stranu) tak sú pravdepodobnosti, že padne guľička do prepážky vľavo alebo vpravo rovnaké, teda $[0, 5; 0, 5]$. V prípade, že je kolík na ľavej spodnej strane trojuholníkovej zarážky, guľičky môžu padať iba vľavo a teda pravdepodobnosti sú $[1; 0]$ (v prípade, že je kolík na pravej strane trojuholníkovej zarážky, to bude naopak).



P5 Losí úřady (pokračování)



InterLoS 2017

Zamysleme se, jak redukovatelná slova mohou vznikat:

1. Skupina délky alespoň 2 složená z jednoho písmene je redukovatelné slovo.
2. Redukovatelné slovo obklopené alespoň jednoprvkovými skupinami stejného písmene je redukovatelné slovo.
3. Zřetězení dvou redukovatelných slov je opět redukovatelné slovo.
4. Zřetězení libovolného počtu stejných písmen, redukovatelného slova, stejného písmene, redukovatelného slova a libovolného počtu stejných písmen (příčemž alespoň jedna z krajních skupin je neprázdná) je opět redukovatelné slovo.

Na základě tohoto pozorování můžeme zkonstruovat naivní rekurzivní algoritmus, který rozhoduje, jestli je slovo redukovatelné. Tento algoritmus zkusí:

1. Odmazat ze začátku i z konce všechna stejná písmena a zkontrolovat, jestli je zbytek slova redukovatelný.
2. Zkusí všechna rozdělení slova na dvě slova a ověří, jestli jsou tato slova redukovatelná.
3. Najde rozdělení odpovídající čtvrtému případu a ověří, že zanořená slova jsou redukovatelná.

Takový algoritmus je korektní, avšak není efektivní – má exponenciální složitost a na vstupech, které byly v zadání, neskončí v rozumném čase. Když se nad ním zamyslíme, zjistíme, že některá rozdělení počítáme opakovaně.

Řešení jsou dvě:

1. Následujeme kroky dynamického programování a začneme napočítávat redukovatelnost pro všechna podslova do tabulky. Tuto tabulku vyplňujeme od nejkratších podslov.
2. Následujeme kroky autora úlohy, který byl líný a prachsprostě přidal do naivního algoritmu cachování pomocí slovníku v Pythonu. A funguje to velmi dobře.

Vzorové řešení je k nalezení v souboru *P5s-solution.py*.

Výsledné heslo: *PPUPPUPUUUPUUUPUUUU*



P6 Wiki (řešení)



InterLoS 2017

Tato úloha šla řešit i ručně, ale tolik času nemáme (na obou stránkách Wikipedie je přibližně 500 odkazů). Možným řešením bylo reprezentovat si stránky a odkazy jako orientovaný graf a použít prohledávání do šířky (BFS, Breadth First Search).

Ve vzorovém řešení je použito prohledávání do hloubky s postupným prohlubováním (Iterative Depth First Search), které je paměťově úsporné. Také si program pamatuje odkazy na jednotlivých stránkách, a proto se nemusí dotazovat víckrát (šetří to čas).

Vzorové řešení je k nalezení v souboru *P6s-solution.py*.

Výsledné heslo: `MOOSEANIMALCARTESIANCOORDINATESYSTEMCOMPUTERPROGRAMMINGCHINAMOOSE`



P7 Batoh (řešení)



InterLoS 2017

V tejto úlohe išlo o *Optimalizačný problém batoha*. Jedným z možných riešení mohlo byť čiastočne náhodné, čiastočne systematické skúšanie vkladania vecí do batohov, avšak na tento problém sa veľmi hodí dynamické programovanie.

Pokiaľ si zdefinujeme kvantitu $w_max(n_items, weight)$ ako maximálnu váhu, prvých n_items vecí, ktorá je $w_max \leq weight$, vieme vypočítať odpoveď rekurzívne a rekurziu následne premeniť na iteráciu. Môžeme si všimnúť, že pokiaľ máme n vecí a kapacita batoha je c , tak hľadáme práve $w_max(n, c)$. Pre konkrétnu odpoveď v úlohe hľadáme, či $w_max(n, c) = c$.

Určite vieme, že $w_max(0, weight) = 0$, pretože so žiadnymi vecami nenaplníme v batohu nič. Ďalej vieme vyjadriť nejaké $w_max(i, weight)$ pomocou $w_max(i', weight')$ kde $i' < i \vee weight' < weight$. Povedzme, že teda chceme toto $w_max(i, weight)$ vyjadriť, nazvime váhu i -tej veci $weight_i$. Ak $weight_i > weight$ tak sa určite $w_max(i, weight) = w_max(i - 1, weight)$, pretože vec určite nemôžeme pridať a teda dosiahnuť väčšiu váhu. Ak ale $weight_i \leq weight$, tak máme:

$$w_max(i, weight) = \max(w_max(i-1, weight), w_max(i-1, weight-weight_i)+weight_i)$$

Tým vypočítame maximum toho, že tam vec i nedáme a ušetríme si miesto v batohu, alebo tam vec i dáme, ale znížime si tým zostávajúce miesto v batohu. Na získanie výsledku počítame $w_max(i, weight)$ pre všetky $i \leq n$ a $weight \leq c$, zdola, aby mali naše predošlé výpočty využitie v tých ďalších.

Riešenie v Pythone nájdete v súbore *P7s-solution.py*.

Výsledné heslo: `0110101011011111`



P8 Mooze (řešení)



InterLoS 2017

Řešení této úlohy je přímočaré – je třeba najít nejkratší cestu. K tomu účelu se nejlépe hodí algoritmus [prohledávání do šířky](#), případně algoritmus [A*](#). Jakmile máme nejkratší cestu, stačí ji projít a posbírat písmenka.

Jedinou komplikací této úlohy může být zadaný formát. Komplikaci s dvojnásobnou šířkou políček je možné elegantně vyřešit předzpracováním, kdy každý druhý sloupec umažeme.

Vzorové řešení naleznete v souboru *P8s-reseni.cpp*.

Výsledné heslo: *YAQVBQHFLFQXPGPPGC*



P9 Transformácia (řešení)



InterLoS 2017

Úlohu šlo riešiť priamočiaro tým, že sa vstupný obrazok načítal a niekoľkokrát sa naňho aplikovali zadané pravidlá. Týmto spôsobom postupne vznikali takéto obrázky:



Prácu si je možné uľahčiť, ak si všimneme, že po prečítaní okolia ako binárne číslo získame priamo index pravidla, ktoré máme aplikovať. Tým sa môžeme vyhnúť zložitému vyhľadávaniu. Možné riešenie nájdete v súbore *P9s-solution.py*.

Výsledné heslo: *SRDCE*



L1 Mastermind sudoku (řešení)



InterLoS 2017

Dle pravidel klasického sudoku lze doplnit hned na začátku 8 čísel, viz první obrázek. Podle druhého řádku navíc vidíme tři čísla, které v tajném kódu nejsou, a tři čísla, které v kódu jsou. U devítky známe i její pozici.

2					3	4	5	9	○ ○
5	9					6		7	● ● ● ● ●
4	6						2		○ ○
7		6						4	● ● ○ ○
1	8		7					6	○ ○ ○
3		9						5	○ ○
6	7						4		● ○ ○ ○
8						9		3	○ ○ ○
9	3				4		6	2	● ● ●

1 ② 3
 ④ ~~5~~ ~~6~~ 9 - - - -
~~7~~ 8 ⑨

Nyní se podíváme na osmý řádek, v koši musí ležet tři čísla z kódu. Zároveň v koši neleží čísla 3, 8 a 9. Tedy naopak v koši musí ležet zbývající trojice, čísla 1, 2 a 4. (O jedničce jsme to dosud nevěděli.)

Nově zjištěnou informaci využijeme při analýze řádku prvního. V koši mají ležet pouze dvě čísla z kódu. Zároveň víme, že v něm leží právě čísla 1, 2, 6, 7 a 8. Jednička a dvojka do kódu patří, tedy osmička nikoliv. Čímž máme informaci o kódu kompletní a dořešení úlohy už nebude těžké.

Ve druhém řádku vystěhujeme osmičku mimo koš a zahrajeme několik tahů klasického sudoku (růžová čísla). Naopak porovnáním devátého a druhého řádku můžeme zahrát hnedou trojku. A tak dále.



L1 Mastermind sudoku (pokračování)



InterLoS 2017

2					3	4	5	9	○ ○
5	9	3				6	8	7	● ● ● ● ●
4	6					3	2	1	○ ○
7		6						4	● ● ○ ○
1	8		7					6	○ ○ ○
3		9						5	○ ○
6	7						4	8	● ○ ○ ○
8						9		3	○ ○ ○
9	3				4		6	2	● ● ●

① ② ③

④ ~~5~~ ~~6~~ 9 3 - - -

~~7~~ 8 ⑨

Kompletně vyplněná tabulka vypadá takto:

2	1	7	6	8	3	4	5	9
5	9	3	1	4	2	6	8	7
4	6	8	5	9	7	3	2	1
7	5	6	9	2	1	8	3	4
1	8	4	7	3	5	2	9	6
3	2	9	4	6	8	7	1	5
6	7	2	3	5	9	1	4	8
8	4	5	2	1	6	9	7	3
9	3	1	8	7	4	5	6	2

Výsledné heslo: 593142687845216973



L2 Dvacetistěn (řešení)



InterLoS 2017

Dvacetisten je třeba složit podle čísel (tedy například jedničky patří k sobě). Jedno z čísel je vždy větší než ostatní. Když se na dvacetisten podíváte tak, abyste viděli na všechny malé číslice jednoho druhu a větší číslo směřovalo vzhůru, čáry stejné barvy vykreslí písmeno. Například jedničky odpovídá písmeno S.

Výsledné heslo: *STROMY*



L3 Miny (řešení)



InterLoS 2017

První kliknutí není ničím ovlivněno. Souřadnice políčka, které se odkryje dalším kliknutím je od kliknutí posunuta doprava o tolik neodkrytých políček, v kolikátém sloupci bylo předchozí políčko a posunuta dolů o tolik neodkrytých políček, v kolikátém řádku bylo předchozí políčko. Při posouvání se přeskovala již odkrytá políčka. Pokud by se měla souřadnice dostat mimo plochu, pokračovala zleva, respektive shora.

Po odhalení všech políček, na kterých nejsou miny, hra zobrazí heslo. Výsledné heslo: *SATELIT*



L4 Dělíme LOSI (řešení)



InterLoS 2017

Během řešení je třeba kombinovat důsledně všechna pravidla úlohy: a) dílky je dovoleno pouze otáčet, b) tučné čáry oddělují dílky stejného tvaru, c) dílky stejného tvaru se nesmí potkat nikde, kde tučná linka není.

Dvě místa, kde lze například začít: Nahoře uprostřed s použitím tučných linek vidíme pět políček, která obsahují stejný tvar dílku (označeno a). Když se podíváme ještě důkladněji, zjistíme, že to nemůže být ani tvar O, ani tvar I, ani tvar S. Zbývá tvar L, který rozmístíme tak, abychom splnili všechny tučné linky a zároveň se nám stejné dílky nepotkaly jinde.

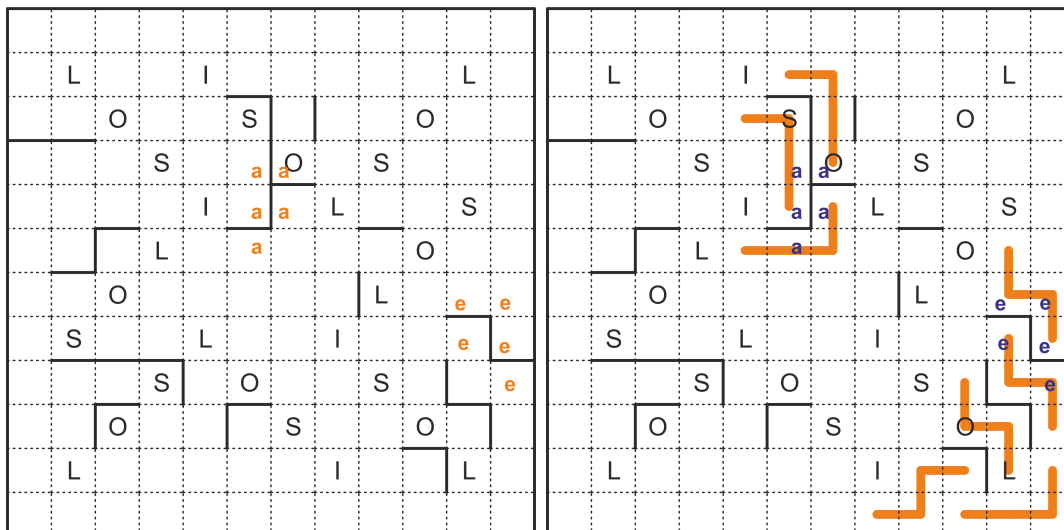
Vpravo dole analýzou tučných linek vidíme opět políčka se stejným tvarem (označeno e). Počínaje horní skupinkou to musí být jedině tvar S, který pak přeneseme i níže. Volné místo v rohu zaplníme jedině dílkem L. A tak dále.



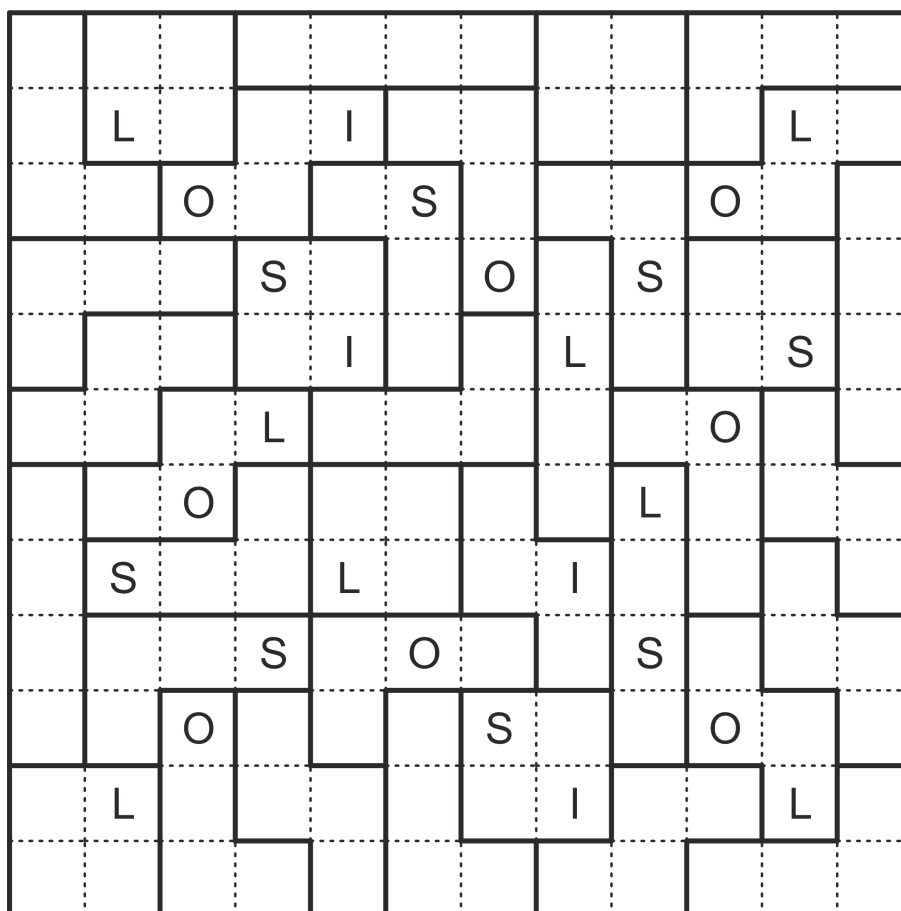
L4 Dělíme LOSI (pokračování)



InterLoS 2017



Kompletní řešení vypadá takto:



Výsledné heslo: *LLOSILLO LLSOLL SSLO ISSLL*



L5 Losí spalničky (řešení)



InterLoS 2017

Řešení šlo snadno nalézt pomocí tabulky, která bude ve sloupcích obsahovat jednotlivá zvířátka a v řádcích klece (nebo naopak).

Nejprve si vyškrtáme, ve kterých klecích která zvířátka být nemohou. Dále víme, že kočka Zelenoočka a myšák Jerry musí být v klecích vedle sebe, čímž umístíme myšáka do klece číslo 5. Kočka tak bude v kleci číslo 4 nebo 6. Aplikováním dalších pravidel, postupným vyškrtáváním a zkoušením, které cesty nevedou ke sporu, se nakonec propracujeme ke správnému řešení.

Výsledné heslo: *KFLEJZABH*



L6 Losí jazyky (řešení)



InterLoS 2017

Všimnout si můžeme nejprve toho, že první velká závorka neobsahuje žádné opakování (\rightarrow), a tedy nám přesně určuje počet písmen ve výsledku. Pro každé písmeno nám pak udává, jakých hodnot může nabývat, a zároveň nám říká, že 2. a 4., respektive 3. a 5. písmeno bude stejné (díky duplikaci, <, pozor na posun způsobený rotací, \rightarrow).

$$\left(\rightarrow \left(< \left((a * c * f * o * r * t) . (c * e * l * p * s * t * z) \right) . (b * j * k * l * q * w) \right) \right) \\ . (e * g * o * p * s * y) . (b * d * f * l * p * v * x) . (b * e * h * i * l * p * t * z)$$

Dostáváme tedy následující varianty pro jednotlivá písmena:

písmeno	1.	2.	3.	4.	5.	6.	7.	8.
možné hodnoty	b, j, k, l, q, w	a, c, f, o, r, t	c, e, l, p, s, t, z	= 2.	= 3.	e, g, o, p, s, y	b, d, f, l, p, v, x	b, e, h, i, l, p, t, z

Další řádek již sice obsahuje opakování, ale jen na jedné straně od určených písmen. Díky rotaci nám určuje, že 3. písmeno je stejné jako 5. a 4. je stejné jako 6., a zároveň nám dává informaci o tom, jakých hodnot může nabývat 3. (a tedy i 5.) písmeno. Dohromady víme, že 2., 4. a 6. písmeno mají stejnou hodnotu, můžeme se tedy podívat, jaké hodnoty jsou dovolené pro 2. a 6. písmeno zároveň. Když to zkombinujeme s tím, co už víme, můžeme vyloučit některé varianty.



L6 Losí jazyky (pokračování)



InterLoS 2017

$$\rightarrow \left(\rightarrow \left(\left((a * b * f * o * q * s * w) \cdot \text{☺} \right) \cdot \text{☺} \right) \right)$$

písmeno	1.	2.	3.	4.	5.	6.	7.	8.
možné hodnoty	b, j, k, l, q, w	o	s	o (= 2.)	s (= 3.)	o (= 2.)	b, d, f, l, p, v, x	b, e, h, i, l, p, t, z

Další řádek popisuje poslední písmeno.

$$\left(\text{☺} \cdot (a * b * e * f * l * q * s * u) \right)$$

písmeno	1.	2.	3.	4.	5.	6.	7.	8.
možné hodnoty	b, j, k, l, q, w	o	s	o (= 2.)	s (= 3.)	o (= 2.)	b, d, f, l, p, v, x	b, e, l

Poslední řádek je opět trochu složitější. Když půjdeme od konce, máme několik možností, kolikrát rozbalit poslední iteraci:

- nulakrát – pak by ale poslední písmeno muselo být jedno z $\{g, m, o, v, z\}$, což ale určitě nemůže být, tuto možnost tedy vyloučíme;
- jedenkrát – toto může být, 7. písmeno by pak bylo v a $(d * i * l * u)$ by mohlo odpovídat jedině prvnímu písmenu (protože nepovoluje ani s ani o);
- dva- a vícekrát – nemůže být, protože bychom neměli žádnou možnost pro předposlední písmeno.

Iteraci lze tedy rozbalit jedině jedenkrát a dostáváme výsledek.

$$\left(\text{☺} \cdot (d * i * l * u) \cdot \text{☺} \cdot (g * m * o * v * z) \cdot \overbrace{(a * e * h * i * j * q * t)}^{\text{☺}} \right)$$

písmeno	1.	2.	3.	4.	5.	6.	7.	8.
možné hodnoty	l	o	s	o (= 2.)	s (= 3.)	o (= 2.)	v	e

A máme hotovo. Výsledné heslo: *LOSOSOVE*.



L7 LOSety (řešení)



InterLoS 2017

Vhodným postupem při třídění kartiček je zaměřit se na některé políčko a spočítat četnost L, O, S. Například u levého horního rohu máme pětkrát L, osmkrát O a pětkrát S. Z toho lze teoreticky sestavit nejvýše 5 setů různých a určitě musí existovat minimálně jeden set, který má vlevo nahoře vždy O. Těchto osm kartiček má na druhém políčku třikrát L, jedenkrát O a čtyřikrát S. Hledaný set s třemi O vlevo nahoře tedy bude mít nahoře uprostřed buďto třikrát L nebo třikrát S nebo tři různá písmena. Mezi takto omezeným počtem možností už není problém správnou variantu najít.

Výsledné heslo: *171721316311154818512146910*



L8 Hodiny (řešení)



InterLoS 2017

Když začneme tím, že jedny hodiny jdou pozpátku, dostaneme osm možností pro to, jaký může být aktuální čas: 5:05, 17:05, 5:05, 9:14, 21:14, 2:15, 14:15, 0:29 a 12:29.

Když si ke každému z těchto časů dopočítáme, kolik by ukazovaly hodiny, které se každou hodinu posunou o 83 minut (jejich čas je v závorce), možné časy se nám zredukuje na tři: 5:05 (6:55), 17:05 (23:31) a 2:15 (2:46). Zbylé časy jsme vyloučili na základě toho, že k nim neexistují v místnosti hodiny, které skáčou o 83 minut. Zároveň můžeme vyloučit čas 5:05, protože čas hodin s 83minutovými skoky by ukazovaly stejné hodiny jako ty, které jdou pozpátku.

Ke zbývajícím dvěma časům (17:05 a 2:15) si dopočítáme čas pro hodiny, které se hýbou jinak v sudou a jinak v lichou hodinu, a protože v místnosti nejsou hodiny, které ukazují čas 2:46, skutečný čas byl 17:05.

Výsledné heslo: *1705*



L9 Matematické lentilky (řešení)



InterLoS 2017

Po prejdení levelu 10 hra vydá heslo.

Výsledné heslo: *RAMPOUCH*



S1 Stříhací (řešení)



InterLoS 2017

Jednotlivým obdélníkům přiřadíme české bankovky. Podle bezpečnostního pásku určíme, jestli se jedná o rub nebo líc. Z orientace nůžek určíme orientaci bankovky – nůžky vždy ukazují hrotem na vrch bankovky. Z vystřižených pozic čteme písmena v pořadí hodnot bankovek.

Výsledné heslo: *POKLAD*





S2 Myslím si číslo (řešení)



InterLoS 2017

Označené vrcholy $\#_1$ až $\#_6$ predstavujú číslice hľadaného čísla. Jednotlivé kružnice naznačujú, v akom poradí sú usporiadané tie isté číslice v rôznych násobkoch hľadaného čísla:

- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 1 = \#_1\#_2\#_3\#_4\#_5\#_6$
- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 2 = \#_3\#_4\#_5\#_6\#_1\#_2$
- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 3 = \#_2\#_3\#_4\#_5\#_6\#_1$
- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 4 = \#_5\#_6\#_1\#_2\#_3\#_4$
- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 5 = \#_6\#_1\#_2\#_3\#_4\#_5$
- $\#_1\#_2\#_3\#_4\#_5\#_6 \times 6 = \#_4\#_5\#_6\#_1\#_2\#_3$

Z týchto vzťahov sa dá odvodiť jediné číslo, ktoré spĺňa všetky podmienky: 142857.

Výsledné heslo: 142857



S3 NyanLos (řešení)



InterLoS 2017

Pro vyřešení úlohy je potřeba najít snímky animace, jejichž kombinace dává dohromady heslo. Na snímek s heslem se odkazují čísla, která se v animaci objevují. Čísla označují, kolikátým následníkem je snímek s částí hesla. Nalezené snímky je třeba překrýt přes sebe, což odhalí heslo.

Kupříkladu snímek s číslem „1“ znamená že snímek s částí hesla je první následník (další snímek). Snímek s číslem „2“ pak druhý následník. Analogicky pro další čísla. Snímek s číslem „0“ pak značí že část hesla se nachází na tomto konkrétním snímku.



Výsledné heslo: MASO



S4 Třípatrová (řešení)



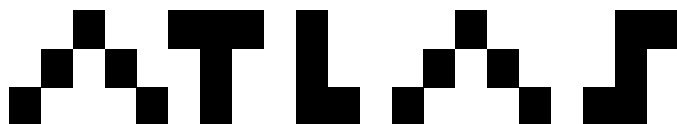
InterLoS 2017

Světlý oblak kolem písmen naznačuje, že v řešení půjde o něco spojeného s počasím, konkrétně skutečně s oblaky. Základní tvary mraků mají své dvoupísmenné kódy a navíc se nachází ve třech výškových patrech.⁵

Vypíšeme sem ty, které se objevují v zadání:

oblaka vysokého patra	
Cirrus	Ci
Cirrostratus	Cs
<hr/>	
oblaka středního patra	
Altostratus	As
Altostratus	As
<hr/>	
oblaka nízkého patra	
Stratocumulus	Sc
Stratus	St
<hr/>	
oblaka přesahující více pater	
Cumulonimbus	Cb

Pokud si jednotlivé mraky postupně vykreslíte podle pater, objeví se písmenka (každý řádek v zadání odpovídá jednomu písmenu v řešení):



Výsledné heslo: *ATLAS*



S5 Futurolosia (řešení)



InterLoS 2017

Primární úlohou v tejto šifre je jednoducho hľadať informácie. Odkaz zo zadania posunie riešiteľa na stránky *Library of Babel*, kde sú uložené záznamy vo forme knižnice.

Každý záznam identifikujú tieto parametre: *hexagon*, *wall*, *shelf*, *volume*, *page*. V priložených poznámkach sú záznamy identifikované číslom (napr. 1:3:1:93), ktoré zodpovedá práve údajom *wall*, *shelf*, *book* a *page*. Samotný záznam potom predstavuje hexagon.

⁵<https://cs.wikipedia.org/wiki/Oblak>



S5 Futurolosia (pokračování)



InterLoS 2017

S těmito údaji je možné najít v knižnici přesně 8 záznamů, které se zhodují s údaji z archívu předcházejících ročníků InterLoSa. Například pro záznam 1:3:1:93:

- do *hex name* zadáme hexagon patriaci k danému záznamu
- *wall* je číslo 1
- *shelf* představuje číslo 3
- *volume* přiřadíme číslo 1
- *page* je číslo 93
- následný text je potřebné pomocí archívu spárovat s konkrétním ročníkem InterLoSa, v tomto případě 2011

Keď všetky ročníky identifikujeme a zoradíme podľa poradia, v ktorom sú uvedené v poznámkach, dostaneme nasledujúci výsledok: *20112012201420162015200920132010*. Ten identifikuje posledný hexagon, kde je potrebné hľadať a ostatné údaje ku konkrétnej stránke sa skrývajú v čase, ktorý futurolos precestoval v pôvodnom zadaní(1:1:20:50). Tieto údaje vedú na stránku, kde prvým slovom je *nit*.

Výsledné heslo: *NIT*



S6 qr_kod.png (řešení)



InterLoS 2017

Při naskenování QR kódu získáte mezitajenku *VIDÍM TO ČERNĚ*, která navádí na využití barev. QR kód je černobílý... anebo ne? Při důkladnějším zkoumání barev jednotlivých čtverečků (nejlépe pomocí nástroje pipeta v nějakém grafickém programu) můžete přijít na to, že ne všechno je čistá černá. Část čtverečků je vlastně hodně tmavě červená, zelená anebo modrá (RGB). Když si pomůžeme pohnutým histogramem, odkryje se nám část tajenky: *HESLO JE TU* – čteme postupně podle schématu RGB. Pokud pohneme na druhou stranu, zjistíme, že ne všechna bílá je vážně bílá a čteme heslo *CIL*.

(Písmeno E je sdílené pro slova JE a HESLO, proto je žluté = červená + zelená.)



S6 qr_kod.png (pokračování)



InterLoS 2017



Výsledné heslo: *CIL*



S7 Počítací (řešení)



InterLoS 2017

Text navádí na využití četnosti písmen v jednotlivých odstavcích. Pokud tyto počty porovnáme s pořadím písmen v abecedě, zjistíme, že v každém odstavci je právě jedno písmeno tolikrát, jaké je jeho pořadí v abecedě. V prvním odstavci si můžeme povšimnout častého výskytu písmene *m*, dále pak pomocí frekvenční analýzy získáme heslo.

Výsledné heslo: *MELOUN*



S8 Plakáty (řešení)



InterLoS 2017

Každý obrazec obsahuje nejfrekventovanější slova nějaké známé knihy (po odstranění vlastních jmen a slov jako jsou předložky, spojky apod., sloučení slov se stejným kořenem a několika dalších úpravách). Na tečkované místo každé knihy patří jméno (některé) z hlavních postav, písmena z těchto jmen čtena v pořadí čísel dávají tajenku.



S8 Plakáty (pokračování)



InterLoS 2017

Použité knihy:

- Alenka v říši divů (Alenka)
- Stopařův průvodce po galaxii (Arthur Dent)
- Obraz Doriana Graye (Dorian Gray)
- Frankenstein neboli moderní Prométheus (Frankenstein)
- Malý princ (Malý princ)
- Mistr a Markétka (Markétka)
- Kniha džunglí (Mauglí)
- 20 000 mil pod mořem (Nemo)
- Robinson Crusoe (Robinson Crusoe)
- Stařec a moře (Stařec)
- 1984 (Winston Smith)

Výsledné heslo: *ANALFABETKA*



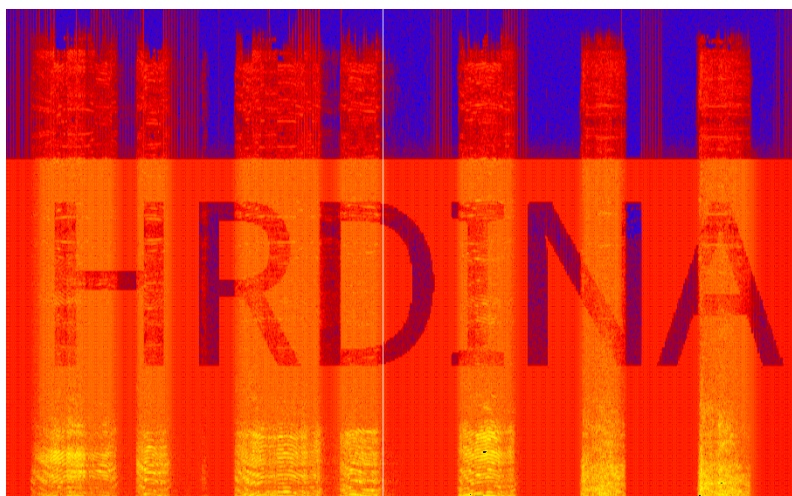
S9 Zvuky (řešení)



InterLoS 2017

Nápoveda úlohy prezradzovala slovo spektrogram, čo znamená že na vyriešenie tejto úlohy bolo potrebné pozrieť sa na spektrogram zvuku. Napríklad pomocou nástroja [sonic-visualizer](#) alebo napríklad [baudline](#), či online nástroja [Academo](#).

Dostaneme tak nasledujúci spektrogram:



Výsledné heslo: *HRDINA*