



P1 Nejtriviálnější (řešení)



InterLoS 2013

Přes úvod k této úloze, který naznačoval volnost, toho bylo velmi mnoho určeno. Pokud se budeme řídit přesně tím, co vyžaduje zadání, dostaneme následující minimalistický program:

- 1) Alloc x set to 2.
- 2) Alloc y set to 2.
- 3) Combine x with y.
- 4) Free x.
- 5) Free y.

Dlužno dodat, že z definice významu příkazů a dle dodatečných omezení vyplynulo, že při splnění veškerých podmínek je výstupní hodnota programu vždy stejná.

správné heslo: 1



P2 Civilizace (řešení)



InterLoS 2013

Chytré řešení: Úlohu nemusíme programovat, stačí si všimnout, že s danými pravidly a vstupem se buňky budou pouze stěhovat nahoru nebo dolů. Díky tomu jich bude na nekonečné rovině po libovolném množství iterací stejný počet. Řešením je tedy počet živých buněk v zadání.

V zadání je celkem $8 \cdot 9 = 72$ pohybujících se objektů tohoto tvaru:

```
--00--  
-----  
-0--0-
```

Celkem je v nekonečném poli $72 \cdot 4 = 288$ živých buněk.

Řešení téměř hrubou silou: Celou úlohu si naprogramujeme a odsimulujeme na dostatečně velké ploše s malým počtem iterací. Uvidíme, že buňky se pouze stěhují. Potom pokračujeme podle prvního řešení.

```
return ( isAlive() && aliveNeighbours() >= 2 && aliveNeighbours() <= 4 )  
        || ( !isAlive() && aliveNeighbours() == 2 );
```

Listing 1: Ukázka funkce, která vrací true, pokud je buňka živá i v následující iteraci.

správné heslo: 288



P3 LoS mistra Leonarda (řešení)



InterLoS 2013

V této úloze bylo třeba naprogramovat výpočet Fibonnaciho posloupnosti, kde každý člen je modulo 2^{24} (tato hodnota odpovídá tomu, že chceme jen poslední 3 bajty příslušného čísla řady).

Pro porovnávání hodnoty n -tého členu se slovem „LoS“ bylo vhodné si toto slovo zakódovat jako číslo (což bylo již v zadání, číslo 0x4c6f53).

Fibonacciho posloupnost ovšem nestačilo počítat pomocí jednoduché rekurzivní funkce pro n -tý člen:

```
int fib( int n ) {  
    if ( n == 1 ) return 0;  
    if ( n == 2 ) return 1;  
    return ( fib( n - 1 ) * fib( n - 2 ) ) % 16777216;  
}
```

Problém této definice je jednak, že se v každém kroku volá 2x rekurzivně (a tedy počet volání je exponenciálně úměrný k n), druhak to, že při postupném výpočtu členů nikdy nevyužijeme to, co je již vypočítané z předchozích členů.

Bylo třeba si najít nebo vymyslet iterativní metodu výpočtu Fibonacciho posloupnosti, asi nejjednodušší je uvědomit si, že každé číslo závisí jen na 2 předchozích a tedy si stačí pamatovat jen tato dvě čísla. Potom je třeba je v každém kroku sečíst a namísto menšího z nich si zapamatovat součet.

```
int a = 0; // pocatecni stav  
int b = 1;  
  
while ( 1 ) { // porad opakuj  
    // zde je mozne vypsati dalsi clen — a  
    int c = (a + b) % 16777216;  
    a = b;  
    b = c;  
}
```

Nyní již bylo třeba jen si počítat kolikáté číslo posloupnosti máme a porovnávat s číslem odpovídajícím slovu „LoS“.

Vzorové řešení v C++ (v příloženém souboru P3-solution.cpp) doběhlo v neměřitelném čase, řešení v interpretovaném Haskellu do 5 vteřin. (Obě řešení byla iterativní – rekurzivní řešení obvykle spadne díky příliš hluboké rekurzi a nejspíš by se stejně nestihlo dopočítat výsledku.)

správné heslo: 7812941



P4 Karpální tunel (řešení)



InterLoS 2013

Ještě než začneme řešit zadaný úkol, učiníme pozorování: pokud ve vstupním ani ve výstupním textu není zastoupeno nějaké písmeno, nebude ho muset Igmar vůbec použít, protože k nalezení nejkratšího řešení by to zřejmě nepomohlo. A protože vstup i výstup sestává pouze z písmen **l**, **o** a **S**, ostatní písmena ze zadané tabulky kláves můžeme vypustit.

Máme tedy textový editor, který je v nějakém stavu, a máme operace, pomocí nichž máme editor dostat do jiného stavu. Stavem editoru přitom máme na mysli:

- textový obsah;
- pozici kurzoru;
- informaci, zda je zmáčkнутý **Shift** (ten totiž může Igmar držet pro zadání několika velkých písmen za sebou);
- informaci, zda editor právě nahrává makro;
- momentální stav registru makra.

Teď si různé takové stavy editoru představíme jako uzly grafu a spojíme tyto uzly (orientovanými) hranami tak, aby z uzlu u do uzlu v vedla hrana, právě když se stiskem jediné klávesy dostane editor ze stavu u do stavu v . Pak je to již jednoduchá úloha nalezení nejkratší cesty v neohodnoceném orientovaném grafu.

K vyřešení použijeme prohledávání do šířky: počáteční stav si zařadíme do fronty, do které budeme postupně, tak jak na ně narazíme, zařazovat všechny další nově objevené stavy. Abychom poznali, že jsme v nějakém stavu již byli (a je tedy zbytečné ho procházet znovu), a abychom věděli, jak jsme se do nějakého stavu nejlépe dostali, budeme si ještě udržovat mapu stavů editoru na jim předcházející stavy uložené spolu s informací, jakou klávesou se z předcházejícího do aktuálního stavu přešlo. Důležitá je zde jediná věc: abychom stavy k dalšímu prozkoumání zařazovali do fronty v tom pořadí, jak jsou abecedně seřazené klávesy vedoucí do nového stavu. Tím snadno naplníme podmínku, abychom v případě několika stejně dlouhých posloupností kláves volili tu, která je nejdříve v abecedě.

Popsaný algoritmus stačí efektivně implementovat a nechat několik vteřin počítat. Vzorové řešení najdete v příloženém souboru `P4-solution.cpp`.

správné heslo: L268SO55O

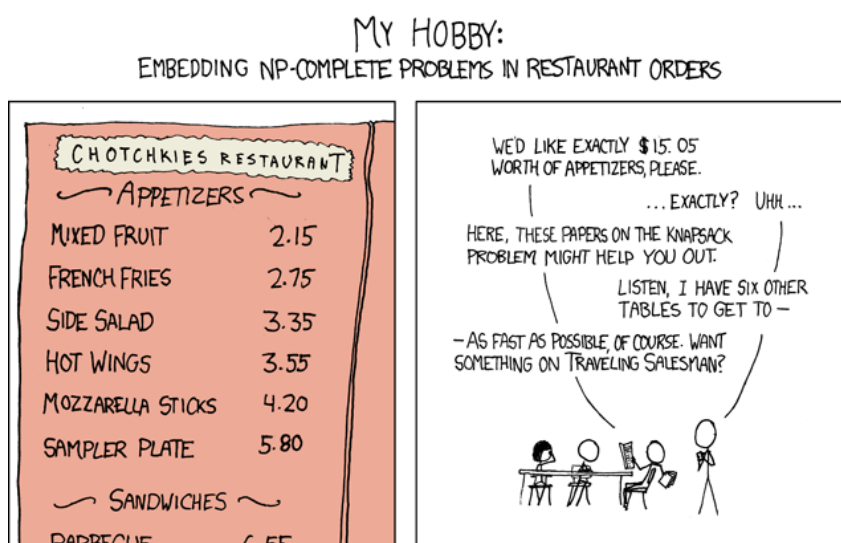


P5 Losí nakupování (řešení)



InterLoS 2013

Tato úloha je typickým příkladem kombinatoricky optimalizačního problému nazvaného **problém batohu** (anglicky *knapsack*). Každý předmět má definovanou „cenu“ (v našem případě prioritu) a „váhu“ (v našem případě cenu). Cílem je nalezení vhodného algoritmu, který dokáže umístit podmnožinu předmětů. Do batohu s omezenou velikostí tak, aby celková „cena“ byla maximální. Předměty jako takové nelze dělit na podčásti (buď v batohu jsou nebo ne), ani je do batohu umístit vícekrát.



Obrázek 1: zdroj: xkcd NP-Complete [<http://xkcd.com/287/>]

Tuto úlohu lze řešit dvěma základními přístupy:

1. Hrubá síla

Tento nejnaivnější (a nejjednodušší) algoritmus postupně prochází všechny možné kombinace jednotlivých předmětů a vybírá z nich takové kombinace, jejichž součet „cen“ je co největší. Výhodou tohoto algoritmu je jednoduchost jeho implementace. **Pro fajnšmekry:** Jelikož prochází všechny možné kombinace (tzn. podmnožiny), je jeho časová náročnost je exponenciální (2^n , kde n je počet předmětů).

2. Dynamické programování

Tento, o poznání složitější, algoritmus je založený na doplňování batohu postupným procházením matice. Nejdříve si připravíme matici, kde sloupce budou reprezentovat velikost batohu od 0 do n a řádky jednotlivé předměty od 0 do m . Algoritmus postupně prochází celou matici od buňky $[0, 0]$ až do $[m, n]$ a v každé buňce rozhoduje, zda-li je lepší vložit do batohu nový předmět a prázdné místo doplnit starými nebo nechat jen ty staré. Po doplnění celé matice je ještě potřeba provést extrakci samotných předmětů. To vyřešíme opětovným projitím matice od posledního řádku a hledáním rozdílů mezi hodnotou předmětů v aktuálním řádku a v řádku o jeden výše.



P5 Losí nakupování (pokračování)



InterLoS 2013

Pro fajnšmekry: Časová náročnost tohoto algoritmu je pak pseudopolynomiální ($O(C \cdot n)$) - tzn. jeho složitost je možné omezit polynomem vzhledem k délce vstupu (n je počet předmětů), ale nikoliv vzhledem k velikosti vstupu (C může být až exponenciálně velké).

Výsledný součet priorit: **361**. Produkty v batohu:

priorita	cena	název
55	10.9	Ano Babiččina zeleninová směs
54	31.9	Tesco Štrúdl
51	22.9	Ananas střední
50	39.9	Mikulášské figurky
43	25.9	Toffifee
40	19.9	Cottage
37	27.9	Pedigree – konzerva
22	21.5	Dobrý hostinec
9	6.9	Croissant 7 days

Ukázková implementace:

Přiložený soubor `P5-knapsack.py` obsahuje okomentovaný kód vzorového řešení využívajícího dynamické programování. Program je napsán v jazyce Python 3 a lze si jej spustit buď na lokálním počítači (Python lze nainstalovat na Windows, Linux i Mac OS) nebo pomocí internetového rozhraní na adrese http://www.compileonline.com/execute_python3_online.php. Soubor `P5-input.txt` obsahuje akční produkty vyextrahované z tabulky v zadání s odpovídající prioritou a cenou z akčního katalogu.

Více informací:

- <http://www.algoritmy.net/article/5521/Batoh> (česky)
- <https://edux.fit.cvut.cz/oppa/MI-PAA/cviceni/MI-PAAdynprog.pdf> (česky)
- http://en.wikipedia.org/wiki/Knapsack_problem (anglicky)
- <http://www.youtube.com/watch?v=EH6h7WA7sDw> (anglicky)

správné heslo: 361



P6 Úloha pro maminku (řešení)



InterLoS 2013

V této úloze bylo potřeba nalézt nejkratší rozvrh splňující podmínky zadání. Jelikož to je dost složitý problém, úloha byla postavená tak, že stačilo rozvrh hledat hrubou silou.

Pro každou z akcí (praní, ždímání, sušení, žehlení) bylo třeba odzkoušet všechna možná pořadí druhů prádla, v nichž se dá akce dělat (pro ty druhy na něž se akce vztahuje). Zároveň však pro každé pořadí dřívějších akcí je potřeba vyzkoušet všechna pořadí dalších akcí (protože to, že jsme vyprali rychle může třeba znamenat, že máme nevhodné pořadí pro žehlení).

To tedy znamená, že pro všechny permutace prádla k praní zkusíme všechny permutace prádla k ždímání a pro všechny tyto kombinace zkusíme všechny permutace sušení a tak dále. . .

Takto si můžeme pro každé pořadí vypočítat délku rozvrhu a pak vybrat tu nejkratší. Vzorovou implementaci najdete v příloženém souboru `P6-solution.hs`

Samotnou délku rozvrhu (pro pevné pořadí prádla u jednotlivých akcí) vypočítáme například postupně po jednotlivých akcích. Ke každému prádlu si budeme držet dobu, kdy na něm doběhla poslední zpracovaná akce a zároveň si budeme držet i celkovou dobu trvání rozvrhu (tedy dobu dokončení časově poslední akce). Tu budeme postupně aktualizovat při zpracování akcí.

Pro každou akci tedy nyní projdeme prádlo v daném pořadí, přičemž si musíme pamatovat, kdy je příslušný přístroj volný. Pro každé prádlo v akci se podíváme, kdy může akce na prádle začít – to je maximum z času, kdy je volný stroj, a času dokončení předchozí akce na prádle. K tomu připočteme délku akce na prádle a uložíme si výsledek jako dobu konce poslední spočítané akce na tomto druhu prádla. U posledního prádla v pořadí (v akci) se navíc musíme podívat, jestli doba ukončení jeho zpracování není vyšší, než aktuální délka rozvrhu, pokud ano hodnotu aktualizujeme.

Takto postupujeme postupně v pořadí akcí, tak jak jsou dané zadáním, a na konci dostaneme délku rozvrhu – pozor to nemusí být čas dožehlení posledního žehleného prádla, prádla k žehlení je docela málo a pokud bylo brzy vyprané, je klidné možné že poslední akce (z hlediska času) je třeba sušení.

Výsledný rozvrh prací měl vypadat následovně (celková délka 340 minut):

	0	80	120	170	200	230	240	260	290	310	315	340
Praní		U	K	R	S			Š				
Ždímání			U		R	S						
Sušení				K		U	R	S				
Žehlení							K	U				

správné heslo: 340



P7 4D bludiště (řešení)



InterLoS 2013

Vzorové řešení (přiložený soubor `P7-solution.cpp`) využívá algoritmu prohledávání do hloubky. Jdeme první cestou co nejdál to půjde. Až dorazíme na konec, tak se vrátíme na poslední odbočku, zase jdeme co nejdál to půjde atd. Cestou sbíráme písmenka do naší tajenky, přičemž si na každé křižovatce zapamatujeme dosavadní stav tajenky. Pokud se vracíme o jednu křižovátku zpět, tak část tajenky od polední křižovátky můžeme zapomenout.

správné heslo: *PAROHY*



P8 Óda na Unicode (řešení)



InterLoS 2013

Po naprogramování dekodujícího programu a jeho spuštěním na poskytnutý soubor byla dekodována následující zpráva:

Åhöj Èrîkû, víš, jak pömhámé s töü naši drúhövöü pröpågàcí pömöcí té ïntérnétövé söütféžé örgànîzövàné někde v Brně... Tåk jsém sî říkáł, že tö tvöjé výslédñé hésłö prö úłöhu P8 sé mî vúbéc ñélíbí, myslím, že bychöm měłî zvölît jîñé. Pöüzîj söüslöví vélmî pödöbñé názvü úłöhy, které má söüvîslöst s Bééthövéñém à Schîllérém. Tåk tö búde ñéjlépší. Zdràví Třømös. P.S. ÅSCII SUCKS!

Souvislost „Ódy na Unicode“, „Beethovena“ a „Schillera“ je Óda na radost, výsledné heslo je tedy „ODANARADOST“.

Vzorové řešení najdete v přiloženém souboru `P8-unicode-solver.c`, generátor zadání pak v souboru `P8-unicode-generator.c`.

správné heslo: *ODANARADOST*



P9 Palindromy (řešení)



InterLoS 2013

Procházíme celý 3D text po znacích. U každého znaku kontrolujeme, zda se po osách x , y a z na dané pozici nachází palindromy. Pokud ano, uchováváme si jejich celkovou délku a středový znak. Nakonec znaky seřadíme vzestupně podle délek palindromů.

Ukázkové řešení naleznete v přiloženém souboru `P9-solution.cpp`.

správné heslo: *LOSOVIPIVOSOL*

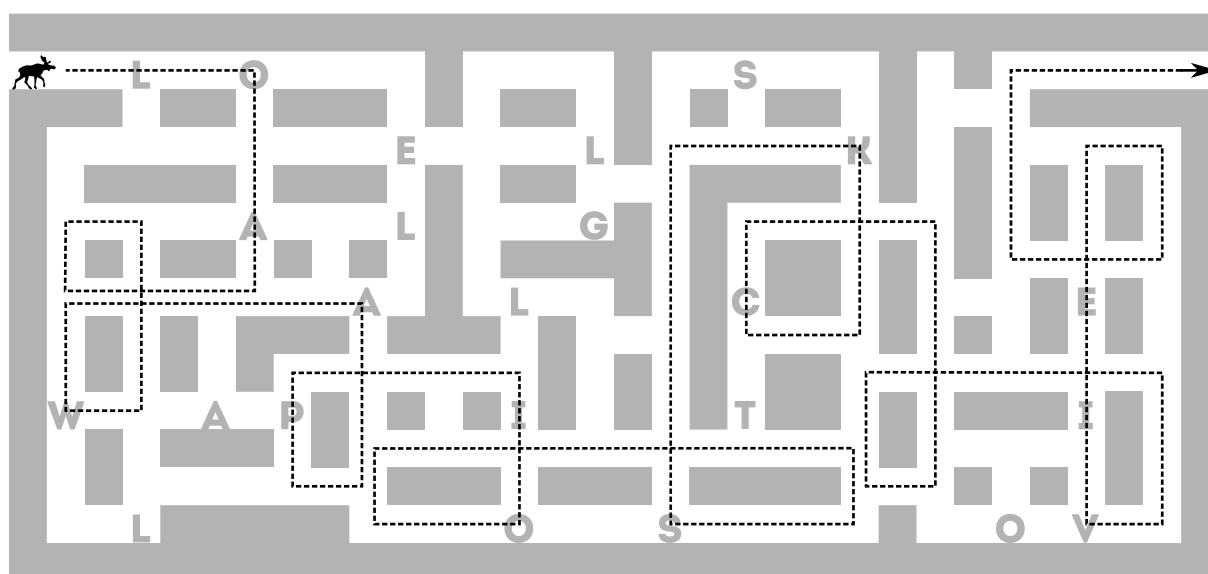


L1 Nothing left (řešení)



InterLoS 2013

Některé odbočky (například hned tu první) nelze na první pohled využít, protože jsou to slepé cesty (všechny odbočky jsou vlevo). Další možný krok je rozdělení bludiště na menší části, ve kterých pak hledáme nejkratší cesty. Například do posledních sedmi sloupců se dá dostat pouze třemi cestami, po bližším průzkumu zůstane dokonce jen jedna. Pokud už nějakou cestu skrz bludiště najdeme, snažíme se ji minimalizovat, jak jen to jde.



správné heslo: LOAWAPIOSKCVIE



L2 MHD (řešení)



InterLoS 2013

Řešení se dá najít jak nad plánem MHD, tak i programem. Procházení do šířky všech možností jízdy z Hlavního nádraží, když si pamatuji, které linky a zastávky již nemohu použít, trvá několik sekund. Z všech nalezených cest již pak stačí jen vybrat tu, která prochází přes Janáčkovo divadlo. Možných posloupností použitých linek je více, ale všechny mají stejnou trasu, a to tuto:

linka	zastávka
-------	----------

–	Hlavní nádraží
4	Malinovského náměstí
11	Janáčkovo divadlo
1	Moravské náměstí
9	Česká
3	Grohova
12	Konečného náměstí

linka	zastávka
-------	----------

25	Čápkova
26	Úvoz
38	Tvrdeho
35	Mendlovo náměstí
5	Poříčí
2	Nemocnice Milosrdných bratří
6	Celní

správné heslo: *HMJMCCKCUTMPNC*



Úlohu je nejjednodušší začít řešit od existujících šipek, jejichž následováním nám vychází, že buňka s číslem 3 ve spodní části tabulky je už „mrtvá“ a tím pádem od ní ve všech směrech získáváme ideální oblasti k dalšímu zkoumání. V dalších krocích je vhodné se soustředit na oblasti mající některé z následujících vlastností:

- v některém z možných směrů leží „mrtvé“ číslo,
- okénko šipky leží blízko rohů tabulky (je omezený počet směrů, kam může ukazovat),
- v některém z možných směrů je buňka obsahující vysoké číslo.

Tímto iterativním postupem lze poměrně rychle doplňovat další a další šipky a tím vytvářet nové zajímavé oblasti. Postupným doplňováním šipek (a s notnou dávkou štěstí), se dopracujete až k výslednému heslu.

[illegible]

správné heslo: 51171347710



L4 Losí pentomino (řešení)



InterLoS 2013

?	U			N	?	N		?	L	?	L
U			?	N							L
U	?					?	Y	Y	Y		
				?				?			I
V	V	?		T	T	?					?
?				T				P	?		I
V		F					?	P	P		I
		?	F	?							?
?			F			?			?		
W	W				X	X	X		Z	Z	?
	W	?				?					Z

správné heslo: UNLLNUYTYVTIVPPFFIWXXZZWX

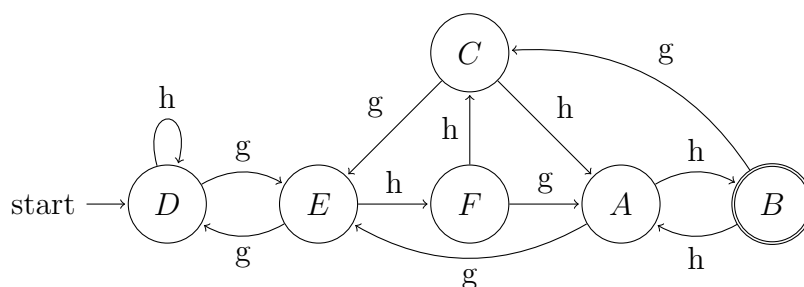


L5 Deterministický les (řešení)



InterLoS 2013

Deterministický les je vlastně deterministický konečný automat, kde jednotlivé stavy tvoří zpěvní ptáci a dva druhy přechodů mezi nimi jsou predátoři. Stačí tedy automat sestavit podle zvukových nahrávek a pak najít nejkratší cestu.



Zvuky *gargamele* a *hrůzostrašnýše* pochází z <http://soundjax.com/>, zvuky ptáků z <http://www.birdsinbackyards.net/birds/featured/Top-40-Bird-Songs>.

správné heslo: DGEHFGAHB



L6 Substituční (řešení)



InterLoS 2013

Díky pevně zadané pozici odmocniny lze okamžitě určit pozici symbolu $=$. Máme vyřešit hodnotu v , takže můžeme hledat vzorec ve tvaru $v = \sqrt{\text{něco}}$. V odmocnině se vyskytují operátory $\div$ a $\cdot$, z čehož vyplývá, že úzávorkování má význam jen pro jmenovatele ve zlomku. Lze si také všimnout často opakujících se symbolů - v se vyskytuje 4x, tedy by mohlo jít o nějaký operátor. v se vyskytuje 2 znaky před koncem odmocniny, mezi kterými se nemůže vyskytovat žádný operátor (intuitivně radí, že by to mělo být násobení, vícenásobné dělení by šlo zjednodušit úpravou výrazu). Nemohou se vedle sebe vyskytovat 2 operátory ani 2 proměnné, musíme zde tedy umístit závorku (ukončení závorky).

Aktuální stav je (podtržené jsou symboly na správné pozici): $\underline{v} = \sqrt{\underline{2} \cdot \underline{g} (\underline{\rho} \underline{S} \underline{m} \cdot)}$

Aby byla zachována sousednost operátorů a proměnných, musí symbol $($ znamenat $\div$ a tedy symbol ρ musí být začátek závorky.

Aktuální stav je: $\underline{v} = \sqrt{\frac{\underline{2} \cdot \underline{g}}{(\underline{S} \cdot \underline{m} \cdot)}}$

Takže už jen hledáme, které veličiny jsou v čitateli a které ve jmenovateli. Jak řešitelé mohli odhadnout, vzorec je fyzikálního rázu. Rozeberme si tedy, jakých jednotek jsou které veličiny (píšu jen ty, které by měl znát i nefyzik).

$$[v] = \text{m} \cdot \text{s}^{-1}$$

$$[g] = \text{m} \cdot \text{s}^{-2}$$

$$[m] = \text{kg}$$

$$[\rho] = \text{kg} \cdot \text{m}^3$$

Gravitační zrychlení g musí být v čitateli, jinak by ve jmenovateli musela být veličina s rozměrem s^4 . Jelikož rychlost nemá mezi jednotkami kilogram, musí se kilogramy v ρ a m vykrátit. Je mnohem logičtější varianta, že rychlost roste s rostoucí hmotností, než aby klesala. Tím získáme vzorec:

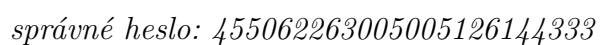
$$v = \sqrt{\frac{2 \cdot m \cdot g}{S \cdot C \cdot \rho}}$$

což je vzorec pro mezní rychlost při pádu. Veličina S je plocha tělesa (projekce plochy ve směru pádu). C je brzdící koeficient, který je závislý na tvaru tělesa. ρ je hustota prostředí. http://en.wikipedia.org/wiki/Terminal_velocity

Dosazení do vzorce je triviální záležitostí, vyřešíte si jej tedy za domácí úkol.



L7 Hrad z kostek (řešení)





L8 Antigravitační kostky (řešení)

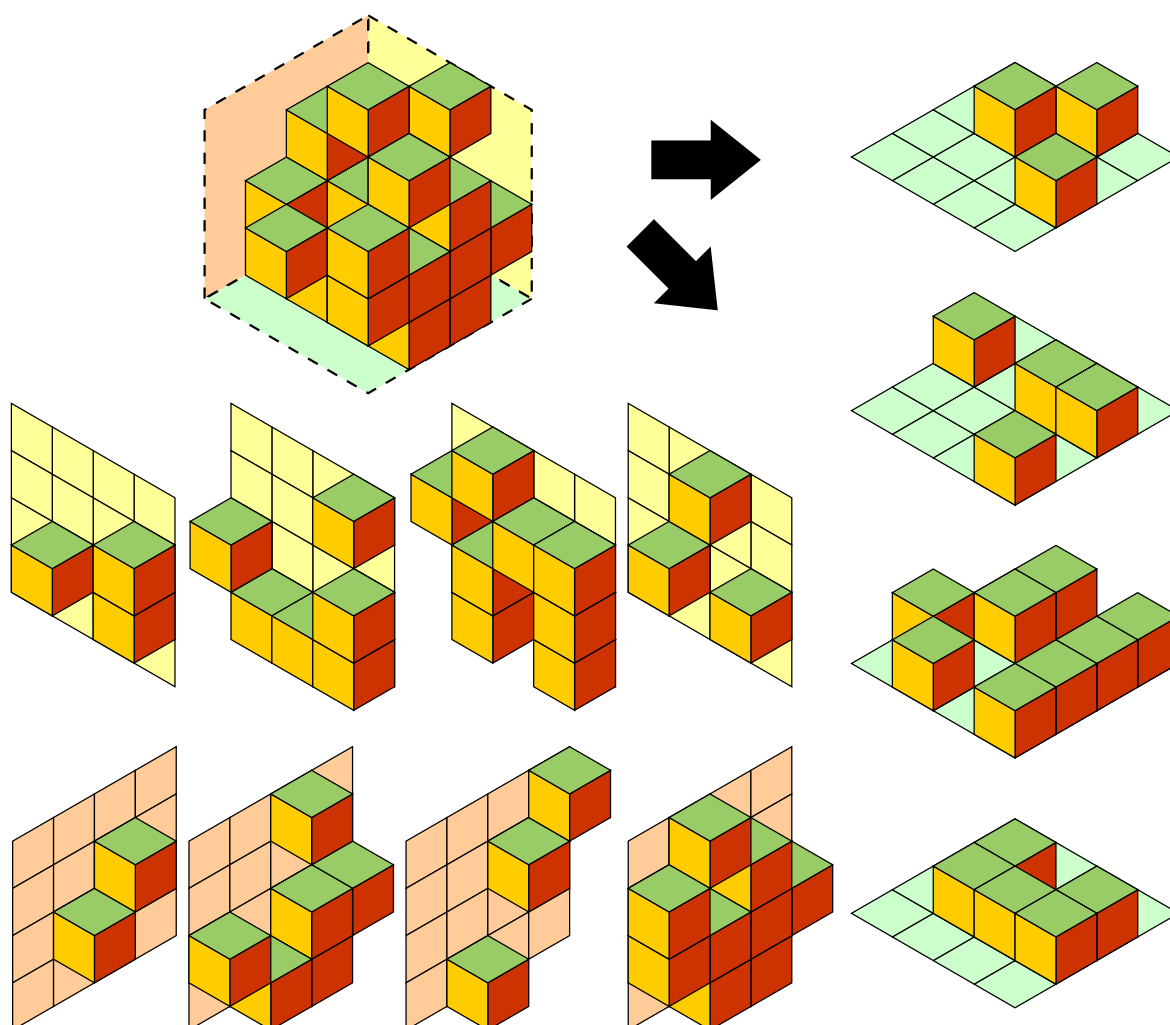


InterLoS 2013

Na začátku se vyplatí prozkoumat stěny kolmé na směr gravitace, tj. žlutou stěnu na první krychli, červenou na druhé a zelenou na třetí. Když je na těchto stěnách místo bez stínu, pak jistě víme, že za ním nikdy není žádná kostka. Tímto způsobem poměrně rychle vyloučíme celkem dost pozic.

Dále je dobré si všimnout, že na jednom místě jsou v řádce čtyři kostky a lze je tedy s jistotou zakreslit. Takto postupně prozkoumáme i ostatní řádky/sloupce a vylučovací metodou umístíme 19 krychlí. Zůstane nám jedno volné místo, na kterém může a nemusí být krychle. Jelikož ale zadání mluví o 20 krychlích, krychli tam umístíme.

Pak už jenom stačí si rozkreslit všechny řezy a spočítat kostky v nich. Úplné řešení i se všemi řezy je vyobrazeno níže.



správné heslo: 233334566889



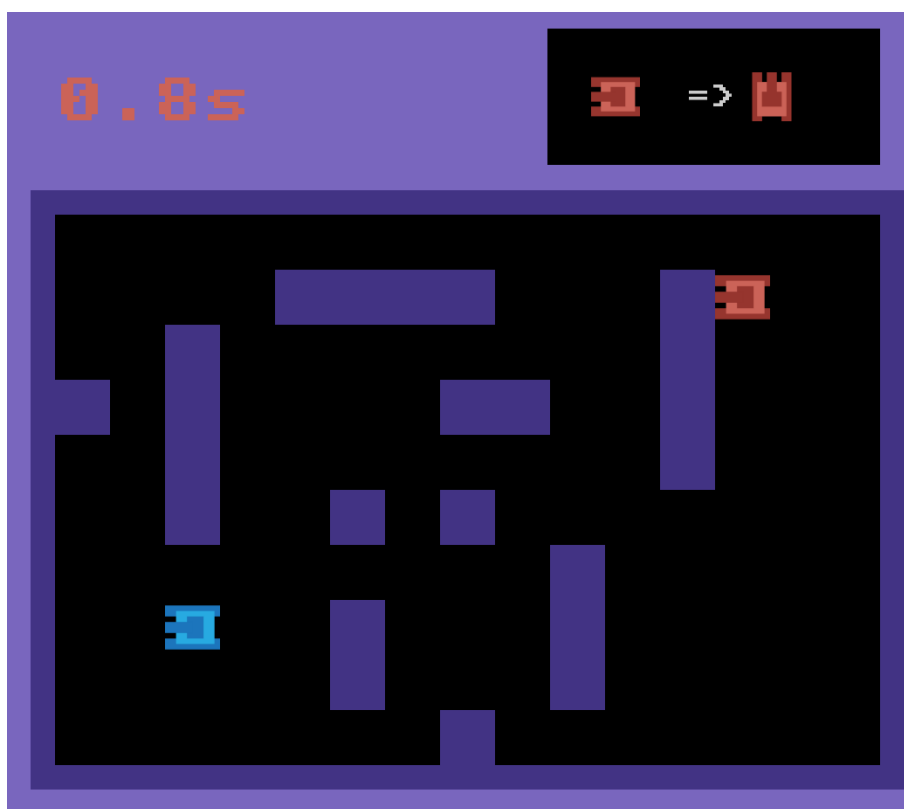
L9 Think or Die (řešení)



InterLoS 2013

Na poskytnuté webstránce byla dostupná onlinová hra pro více hráčů. Každý tým po přihlášení vyčkal na trojici dalších týmů a poté začala DOSově vypadající střílečka, ve které vždy dva týmy ovládali jeden tank. Jedno kolo ovládal jeden tým pohyb tanku, druhý tým pak pohyb děla a střelbu. V příštím kole se ovládání prohodilo. Na rozmyšlení tahu jste měli 3 sekundy, během kterých jste museli domyslet, co asi bude chtít dělat ten druhý... jinak jste se mohli rozloučit se svým tankem. . .

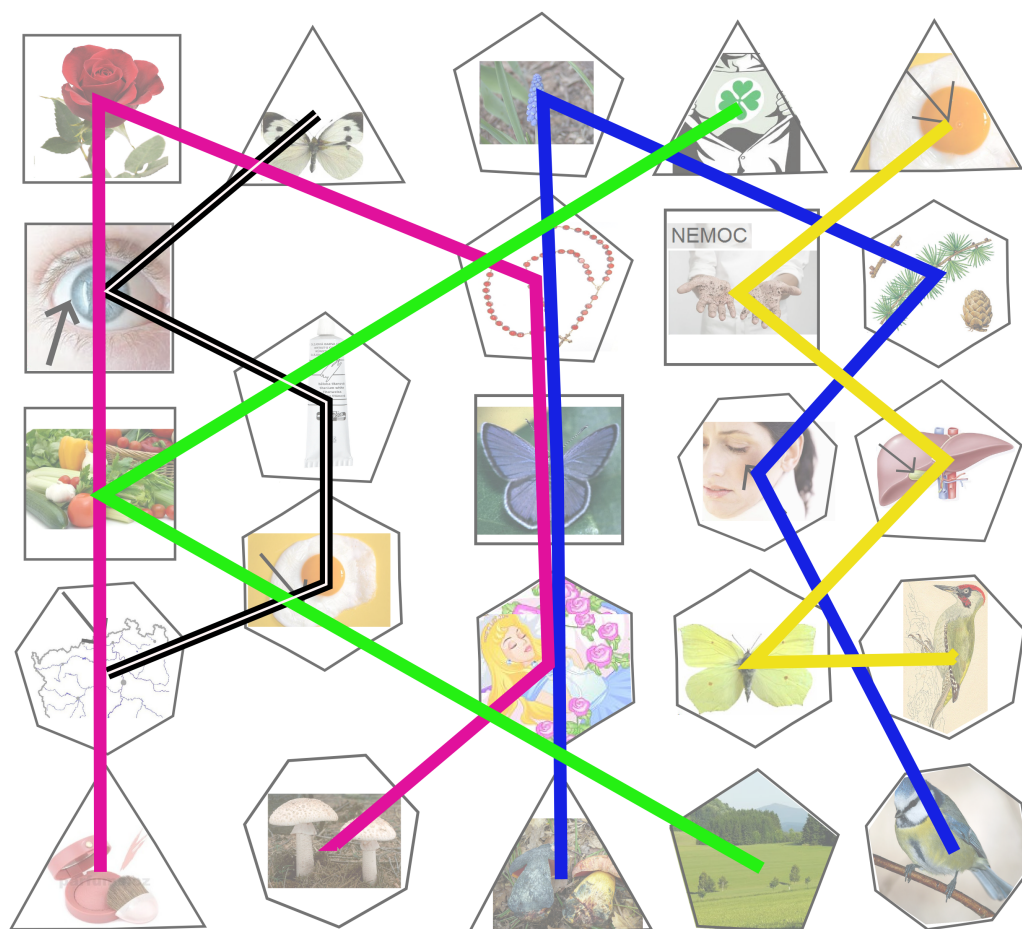
Cílem hry bylo zlikvidovat nepřátelský tank vystřelením na něj, tak, aby ve stejném kole nevystřelil na vás, protože v takovém případě by nastala remíza. Týmy, které remizovali nebo prohráli mohli hrát znovu. V případě, že se vám zásah zdařil, obdrželi jste heslo, které znělo „PAPRIKA“.



správné heslo: PAPRIKA



žlutá: žloutek, žloutenka, žlučník, žlutásek, žluna



správné heslo: SRDCE



S2 Čísla (řešení)



InterLoS 2013

V zadání jsou pouze česká poštovní směrovací čísla. Po přiřazení pošty každému PSČ můžeme z prvních písmen obcí číst mezitajenku „PLANZNINAJITAKORATSUDAPSC“. Vezmeme tedy sudá PSČ a zobrazíme je na mapě. Dávají tvar známého souhvězdí, heslo je tedy „VELKYVUZ“.

798 48	P rotivanov	468 43	A lbrechtice v Jizerských horách
270 61	L ány	582 92	K amenná Lhota
362 35	A lbertamy	582 31	O krouhlice
691 64	N osislav	592 33	R adešínská Svatka
285 21	Z braslavice	742 55	A lbrechtický
277 51	N elahozeves	588 56	T elč
683 23	I vanovice na Hané	684 01	S lavkov u Brna
357 34	N ové Sedo u Lokte	783 91	U ničov
735 43	A lbrechtice u Českého Těšína	691 83	D rnholec
252 42	J esenice u Prahy	352 01	A š
664 91	I vančice	273 76	P ozdeň
675 79	T asov	392 01	S oběslav
		561 61	Č ervená Voda

Mapa se zaznačenými obcemi je k nahlédnutí zde:

<https://mapsengine.google.com/map/edit?mid=zSHVU4ECXnRo.kyh8fSlV1T6A>

správné heslo: VELKYVUZ



S3 Malé měkké slovo (řešení)



InterLoS 2013

Úloha využívá známé klávesové zkratky v programu *Microsoft Word* – což upřesňuje název úlohy. Každá vytučeněná část odkazuje na jednu klávesovou zkratku (Ctrl + nějaký znak), nicméně úloha je řešitelná i *OpenOffice* a jeho derivátech, případně s pomocí online dostupnému seznamu klávesových zkratk.

To byl ale den. Kéž bych některé chvíle mohl zkrátit a užívat si příjemnější kratochvíle.

*Tak nejprve jsem na křižovatce zahnul **doprava** (Ctrl+R), i když tam byl zákaz vjezdu. Samozřejmě, že mě hned chytil policajt, napařil mi pokutu, přičemž částku 1000 Kč ve svém bločku dvakrát **podtrhnul** (Ctrl+U)! Vztekly jsem se **vracel zpět** (Ctrl+Z) a zastavil u nejbližšího automatu na kávu. **Otevřel** (Ctrl+O) jsem peněženku a zjistil jsem, že mám posledních pár drobných, ach jo. **Vložil** (Ctrl+V) jsem je do automatu, načej se na displeji objevil nápis: porucha. Zavola jsem do servisu, aby tu ztracenou mašinu **otevřeli** (Ctrl+O) a zachránili moje mince. Za ně jsem si pak raději místo kávy koupil **tučnou** (Ctrl+B) klobásku. Odtud jsem pospíchal do **kurzu italštiny** (Ctrl+I). I když to nebylo daleko a jen stačilo odbočit **doleva** (Ctrl+L), měl jsem pětiminutové zpoždění, co už. Svě učebnice a úkol jsem si pěkně **zarovnal na střed** (Ctrl+E) lavice a tvářil se jako by nic.*

*Když jsem se konečně večer **vrátil** (Ctrl+Z) domů, rekapituloval jsem svůj zpackaný den. Koukal jsem na pokutu s dvakrát **podtrženou** (Ctrl+U) tisícovkou. Kdybych tak mohl udělat **krok zpět** (Ctrl+Z)! Rozhodl jsem se, že svůj život změním a na ledničku jsem si napsal a barevně **podtrhnul** (Ctrl+U) své nové předsevzetí: vždy vše raději zkontroluju.*

Přečtením písmen tvořících příslušné klávesové zkratky, dostaneme správu „RUZOVOBILE ZUZU“. Heslem je tedy anglický název oblíbené sladké pochoutky.

správné heslo: MARSHMALLOW



S4 Sociálně datová krychle (řešení)



InterLoS 2013

Po prvním prozkoumání šifry je jasné, že půjde o pojmenování bodů v grafu. Z podstaty dat, která grafy znázorňují, se dá odtušit, že půjde pravděpodobně o země.

Dalším krokem je tedy určení zdroje dat, tedy nalezení něčeho, co by prozrazovalo, že přes 500 kin, více než 21 000 zubařů a o trochu víc než 0 dnů potřebných k odstartování podnikání je právě v Austrálii. No a právě k tomu pomůže nápověda pod grafy „Powered by UN Data“, neboť prostým googlením „UN Data“ prakticky ihned dojdete na stránky <http://data.un.org/>, kde už vše snadno dohledáte.

Správné přiřazení dat k zemím vypadá takto:

Time required to start a business in days (2012)	Number of dentistry personnel	Number of cinemas (2005)	země
2	21296	519	Australia
17	15	5	Niger
7	4 126	241	Norway
25	4 467	179	Austria
66	113	4	Namibia
8	2 431	21	Azerbaijan

Minimum legal age for marriage without consent – females (years)	Currently married men and women – married women in the age of 15 – 19 years (percentage), latest estimate	Maternity leave benefits – Length of maternity leave (weeks)	země
20	0.6	14	Japan
18	23.4	8	Malawi
18	20	12	Ecuador
18	0.2	52	Norway
18	0.6	28	Australia

Pak už stačí jen přechíst první písmena doplněných zemí ve správném pořadí (určeném spojnici mezi jednotlivými body) a vyhledat, obě „jména annana“, tj. bývalého generálního tajemníka OSN. Snadno zjistíte, že se jmenuje *Kofi Atta Annan*.

správné heslo: KOFIATTA



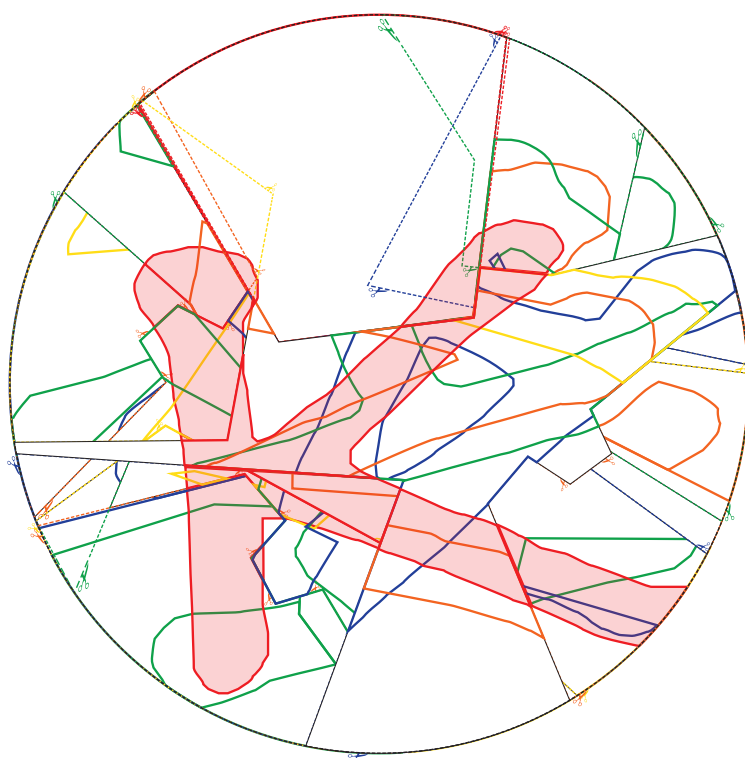
S5 Mandala (řešení)



InterLoS 2013

Jednotlivé díly je potřeba správně vystříhat, poskládat na sebe a vždy pro každý zub pootáčet tak, aby tento zub byl na všech dílech na stejném místě a byly vidět všechny křivky dané barvy. Pro každý zub nám mandala vykreslí jedno písmenko. Tato písmenka seřadíme podle duhy a vyjde kód „KNIHA“.

První písmeno:



správné heslo: KNIHA



S7 Morse (řešení)



InterLoS 2013

Když přímo ze zadání přečteme morseovku, vyjde nám „TAK NE“. Důležitá jsou pozorování, že máme oddělovač i na začátku a na konci a že mezery mezi jednotlivými tečkami, čárkami a oddělovači mají různé délky. Stačí tedy kratší mezeru interpretovat jako tečku, delší jako čárku a při zachování oddělovačů máme kód „MUZEUM“.

správné heslo: MUZEUM



S8 Troubení (řešení)



InterLoS 2013

Stejně jako je troubení zvukovým projevem losů, stejně tak je zpěv písní zvukovým projevem člověka. Obrázky na každém řádku asociují název nebo začátek písničky. Šedé čárky určující, které slovo z názvu nás zajímá, číslo pak písmeno, které nás zajímá. Přečtením vyznačených písmen získáte text „HESLO JE HROUDA“.



Jahody mražený

Řekni, *kde* ty kytky jsou

Nosorožec

Kočka *leze* dírou

Money, Money, *Money*

Kde domov *můj*

Tři *kříže*

Bedna (v)od *whisky*

Tři čuníci

*Trezo*r

Karavana *mraků*

*Yester*day

Nad Tatrou sa *blýska*

správné heslo: HROUDA

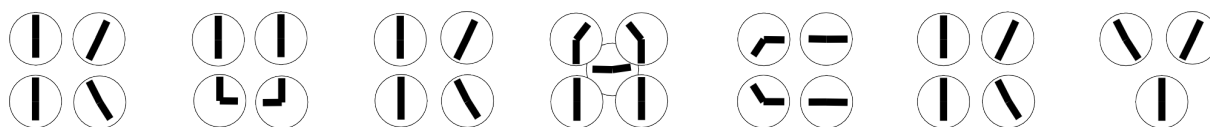


S9 Čas (řešení)



InterLoS 2013

Pokud si dané digitální časy, překreslíte do analogových, tak vám spolu ručičky hodin utvoří text, který je zároveň i heslem.



správné heslo: KUKACKY