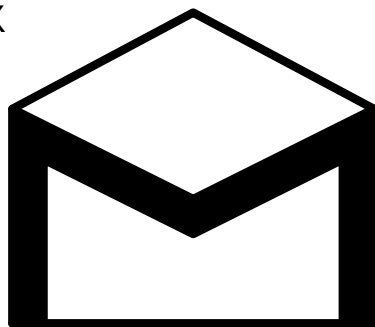
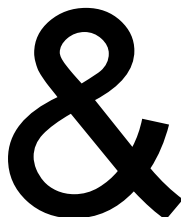
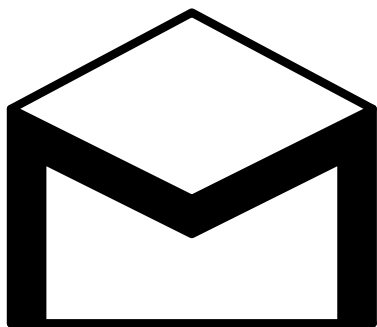


STUDENTSKÝ ČASOPIS A KORESPONDENČNÍ SEMINÁŘ

Ročník XXX

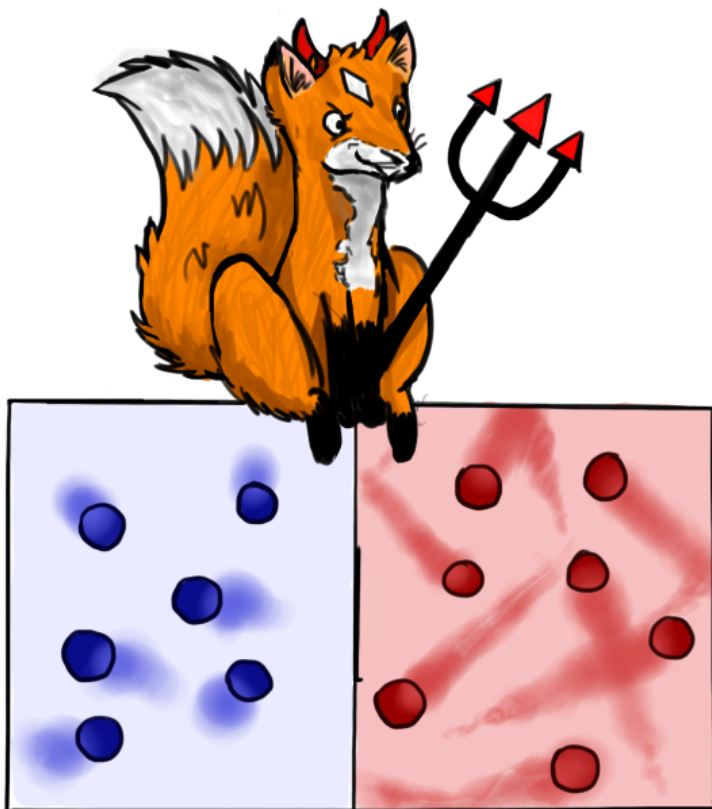
Číslo 3



MATEMATIKA

FYZIKA

INFORMATIKA



Uvnitř najdete několik témat a s nimi souvisejících úloh. Zamyslete se nad nimi a pošlete nám svá řešení. My vám je opravíme a ta nejzajímavější z nich otiskneme. Nejlepší řešitelé zveme na podzim a na jaře na soustředění.



Milí řešitelé,

tímto číslem se oficiálně dostáváme do poloviny 30. ročníku!

V témátku *Termodynamika* se tentokrát zaměříme na entalpii. Že nevíte, co to je? Nasměrujte svoji energii správným směrem, tedy k *Termodynamice*, a přečtěte si to!

Druhé témátko, *Lean*, se konečně dostává k hlavní přednosti tohoto jazyka, kterou je interaktivní dokazování matematických tvrzení. Čeká vás hromada rovností a nerovností, a také se dozvíte, že když jste nějaké tvrzení zatím nedokázali, tak pokud se Leanu hezky omluvíte, tak kód stejně zkompileje – *sorry*.

Hex v tomto dílu připomene problémy, které jsou stále otevřené, abyste se nad nimi mohli znovu zamyslet s novými znalostmi, které získáte při čtení řešení. Otiskujeme výběr hned několika z těch, která jste nám zaslali, spolu s důkazem, že v základní hře má vyhrávající strategii červený hráč; a konkrétní vítěznou strategií pro 7×7 od Dr.^{MM} Martina Těšitele.

Poslední témátko, *Genetika*, tentokrát nic nového nepřináší, můžete ale stále řešit problémy zadané v minulém čísle. Nebo samozřejmě můžete, tak jako ke všem témátkům, napsat článek.

To ale není všechno! Kromě tématků tentokrát otiskujeme hned tři články od řešitelů. První z nich zaslal Dr.^{MM} Martin Těšitel k témátku *Termodynamika* a věnuje se v něm tvorbě baterie z bublinkové limonády. Zvládnete přimět lahev s Kofolou svítit? Přečtěte si Martinův článek a můžete to zkusit! Druhý článek jménem *Axiomatizace přirozených čísel* od Doc.^{MM} Jany Uglickich je o poznání teoretičtější. Pokud vás láká smršť důkazů, na jejímž konci budete umět sčítat a násobit přirozená čísla, tak se můžete hned pustit do čtení. Třetí článek s názvem *Problém klikujícího hráče výzvy* nám zaslal Mgr.^{MM} Jáchym Löwenhöffer a též jej doporučujeme vaší pozornosti. Získané poznatky se vám budou jistě hodit v praktickém životě, a to především v případě, že hrajete „výzvu“ a „Dej si deset!“

Mnoho vánoční nálady přeji

Vaši organizátoři



Obsah

Téma 1 – Termodynamika	3
Řešitelský článek – Limonádovo-bublínková baterie.....	13
Téma 2 – Programování a dokazování v Leanu	15
Řešitelský článek – Axiomatizace přirozených čísel	28
Téma 3 – Hex	35
Téma 4 – Genetika	44
Řešitelský článek – Problém klikujícího hráče výzvy	45

Zadání témat

1. deadline: 2. ledna 2024 | 2. deadline: ~~30. ledna~~ 6. února 2024

Téma 1 – Termodynamika

Vítejte u dalšího dílu tématka. V tomto díle najdete nová zadání, řešení předchozího dílu včetně řešení Problému 2.3 od Dr.^{MM} Lucie Zůnové. Také zde naleznete článek od Dr.^{MM} Martina Těšitele, který byl inspirován prvním dílem a popsal hypotetickou limonádovou-bublínkovou baterii.

Problémy 2.1, 2.4 a 2.5 o veličinách závislých na teplotě a kalorimetrech necháváme stále otevřené. Jejich znění připomínáme zde:

Problém 2.1: *Najděte nějakou veličinu závislou na teplotě a její hodnoty pro vámi vybrané materiály. Vysvětlete, s jakými vlastnostmi si tuto veličinu asociujeme. Kdy je smysluplné její závislost zanedbat? A kdy to naopak provést nemůžeme? Návod: můžete také popsat mechanismus fungování přístrojů, které mají za úkol regulovat a/nebo zaznamenávat teplotu. Vynechejte ze svých řešení veličiny zvané měrná tepelná kapacita a tepelná kapacita, tu zná každý ze školy a navíc je o ní další příklad :).*



Problém 2.4: *Popište detailněji, jak fungují moderní kalorimetry a jejich výhody a limity. Jak vypadá vzorek, který je ideální adept na měření kalorimetrem? Jaké vlastnosti by vzorek mít neměl?*



Problém 2.5: *Zkuste experimentálně změřit hodnotu měrné tepelné kapacity. Může vám na to posloužit jednoduchý model kalorimetru zvaný Blackův kalorimetr. V principu jde o to, že z měření určíte, kolik tepla se vymění mezi ledovou nádobou (to odpovídá množství roztáté vody) a tělesem. Budou se vám lépe měřit*



látky s vysokou nebo nízkou měrnou tepelnou kapacitou? Zkuste zdůvodnit. Pokud najdete jiný způsob měření c_V , popište jej a vyzkoušejte jej.

Vzorová řešení 2. dílu

Úloha 2.2

Zadání:

Pro výpočet měrné tepelné kapacity neznámé látky v závislosti na teplotě lze využít vzorec $c_V = aT^2 + bT + c$, kde

- $c = 28,87 \text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}$
- $b = 14,59 \cdot 10^{-3} \text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-2}$,
- $a = -1,7 \cdot 10^{-6} \text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-3}$.

Předpokládejme, že v rozsahu zadaných teplot měření nedochází ke změně skupenství.

1. Jak velké množství tepla odevzdá těleso, pokud se bude ochlazovat z teploty 90°C na teplotu 10°C ?
2. Jak velké množství tepla odevzdá těleso, pokud se bude ochlazovat z teploty 50°C na teplotu 10°C ?
3. Uvažujte bod 1 s tím, že ve výpočtech použijete střední hodnotu měrné tepelné kapacity. Tu vypočtete jako průměr z hodnot c_V pro okrajové teploty. Uveďte její hodnotu.
4. Porovnejte jednotlivé výsledky. Proč výsledek bodu 2 není polovina hodnoty výsledku bodu 1? Liší se hodnoty v odpovědích bodů 1 a 3? Proč se liší/neliší? Pokud se liší, napište, která z nich je větší a proč.

Řešení:

Nejprve vyřešíme bod 1. Když známe měrnou tepelnou kapacitu c_V , počáteční teplotu T_1 a koncovou teplotu T_2 , můžeme odevzdané teplo Q vypočítat ze vztahu $Q = \int_{T_1}^{T_2} c_V(T)m \, dT$, kde m je hmotnost tělesa.

Jak na tento vztah přijdeme? Funkce pro měrnou tepelnou kapacitu $c_V(T)$ popisuje, kolik tepla je potřeba na ohřátí tělesa o hmotnosti 1 kg o 1 K . Protože její velikost však závisí na teplotě, nejde jen dosadit hodnoty teplot dvou stavů, ale musíme integrovat. Integrace v tomto případě znamená, že pro každou hodnotu T ohřev o malý rozdíl teplot dT vyžaduje dodání tepla $dQ = c_V(T)m \, dT$. Ohřev z T_1 na T_2 získáme sečtením všech velmi malých (přesněji infinitezimálně malých) dílků dQ , čemuž odpovídá teplo $Q = \int_{T_1}^{T_2} dQ = \int_{T_1}^{T_2} (aT^2 + bT + c)dT$.

Hmotnost tělesa zde neznáme, nicméně můžeme spočítat odevzdané teplo vztažené na jednotkovou hmotnost tělesa, tedy $\frac{Q}{m}$. I tento údaj je odpovědí na položenou otázku (arbitrárně si můžeme velikost hmotnosti zvolit jako např. 1 kg). V zadání jsou teploty uvedeny ve °C, ale uvedený vztah pracuje s termodynamickou teplotou, takže musíme tyto hodnoty převést na kelviny, tj. $T_1 = 363,15\text{ K}$ a $T_2 = 283,15\text{ K}$.

Po dosazení vztahu pro měrnou tepelnou kapacitu ze zadání do integrálu dostaneme:

$$\frac{Q}{m} = \int_{T_1}^{T_2} (aT^2 + bT + c) dT = \left[\frac{1}{3}aT^3 + \frac{1}{2}bT^2 + cT \right]_{T_1}^{T_2} \doteq -2672,506\text{ J} \cdot \text{kg}^{-1}$$

Těleso tedy odevzdá teplo $2672,506\text{ J} \cdot \text{kg}^{-1}$. Že se jedná o teplo vztažené na jednotkovou hmotnost, je vidět i z jednotek. To, že teplo je odevzdané, vidíme z jeho záporné hodnoty při výpočtu integrálu.

V bodě 2 bude celý výpočet probíhat stejně, změní se jen meze integrálu. Tentokrát tedy budeme mít $T_1 = 323,15\text{ K}$ a $T_2 = 283,15\text{ K}$. Po dosazení těchto hodnot zjistíme, že těleso odevzdá měrné teplo $1325,460\text{ J} \cdot \text{kg}^{-1}$. Z jednotek vidíme, že se opět jedná o teplo vztažené na jednotkovou hmotnost.

Nyní vyřešme bod 3. Nejprve vypočítáme střední hodnotu měrné tepelné kapacity $\overline{c_V}$ jako průměr z hodnot $c_V(T_1) \doteq 33,944\text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}$ a $c_V(T_2) \doteq 32,865\text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}$. Tyto dvě hodnoty byly dosazeny do vztahu pro c_V v zadání. Průměrná hodnota je tedy

$$\overline{c_V} = \frac{1}{2} (c_V(T_1) + c_V(T_2)) \doteq 33,405\text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}$$

Integrál se nám použitím $\overline{c_V}$ zjednoduší, protože již nebudeme integrovat funkci závislou na teplotě, ale pouze konstantu:

$$\frac{Q}{m} = \int_{T_1}^{T_2} \overline{c_V} dT = [\overline{c_V} T]_{T_1}^{T_2} \doteq -2672,361\text{ J} \cdot \text{kg}^{-1}$$

Nakonec porovnejme jednotlivé výsledky. Výsledek bodu 2 není polovinou výsledku bodu 1, protože se nejedná o lineární závislost. Pokud by vztah pro $\frac{Q}{m}$ byl lineárně závislý na teplotě (což by platilo pro konstantní hodnoty c_V nezávislé na teplotě), potom by při polovičním rozdílu teplot skutečně bylo odevzdáno poloviční teplo. Protože však c_V není konstantní, ale závisí na teplotě kvadraticky, nemůžeme očekávat, že se odevzdané teplo při polovičním rozdílu teplot zmenší právě dvakrát. Jelikož koeficienty a a b jsou vzhledem k velikosti koeficientu c velmi malé, nebude se však výsledek bodu 2 od poloviny výsledku bodu 1 lišit moc.

Dr.^{MM} Lucie Zůnová zmiňuje, že s rostoucí teplotou bude hodnota c_V stoupat a na ochlazení nebo ohřátí o jeden stupeň Celsia bude potřeba odebrat nebo dodat větší a větší množství energie.

Podobně se výsledky bodů 1 a 3 musí lišit opět kvůli nelinearitě závislosti měrného tepla $\frac{Q}{m}$ na teplotě. Liší se o velmi malé jednotky promile, což odůvodňuje to, že ve spoustě středoškolských učebnicích je měrná tepelná kapacita nebo molární tepelná kapacita uváděna jako konstanta nezávisající na teplotě. Občas se vyplatí ve fyzice zanedbávat věci a zjednodušit si tím své fyzikální problémy :)

Problém 2.3

Zadání:

Porovnejte měrné tepelné kapacity vodičů/nevodičů, pevných/kapalných/plynných látek. Co vám hodnoty říkají o jejich vlastnostech? Jsou dobré tepelné vodiče i dobré vodiče elektrického proudu? Proč tomu tak asi je?

Tento problém hezky a srozumitelně zpracovala Dr.^{MM} Lucie Zůnová. Přikládáme její řešení. Text je srozumitelný, obsahuje pěkná vysvětlení toho, co pozorovala v datech, která si našla.

Řešení:

(normální podmínky)	$J \cdot kg^{-1} \cdot K^{-1}$	skupenství	je dobrý vodič el. proudu?	je dobrý tepelný vodič?	poznámky
zlato	129	s	ano	ano	kov
stříbro	235	s	ano	ano	kov
měď	383	s	ano	ano	kov
diamant	509	s	ne	ano	krystalická
sklo	840	s	ne	ne	amorfní
beton	880	s	ne	ne	amorfní
kyslík	918	g	ne	ne	
vzduch	1012	g	ne	ne	
dřevo	~ 1700	s	ne	ne	amorfní
benzin	2220	l	ne	ne	
polyethylen	2302	s	ne	ne	amorfní
ethanol	2440	l	ne	ne	
voda	4180	l	ne	ne	
helium	5193	g	ne	ne	

Měrná tepelná kapacita u látek určuje, kolik energie je potřeba na jejich ohřátí o jeden teplotní stupeň na jednu jednotku hmotnosti – tedy jak snadno je lze ohřát a jak rychle toto získané teplo budou držet. Z tabulky vidíme, že ani kapaliny, ani plyny nejsou dobrými vodiči jak tepla, tak elektrického proudu. To je způsobeno molekulární strukturou – v kapalinách a plynech nedochází k častým interakcím molekul, tudíž nemohou vést tak dobře jako pevné látky. Je tedy logické, že jejich měrná tepelná kapacita bude vysoká. U plynů je nižší, protože rychle se pohybující vysokoenergetické částice mají tendenci si předávat teplo lépe než pomalejší molekuly v kapalinách.

Naopak nejlepšími vodiči jsou kovy – jejich kovová struktura s volnými elektrony umožňuje přenos proudu i tepla velmi efektivně. Tyto vlastnosti spolu úzce souvisí, protože obě závisí na podobných podmínkách – vhodném molekulárním uspořádání. Proto podobné vlastnosti nevidíme u amorfních látek. Jejich měrná tepelná kapacita není tak vysoká jako u plynů, to je dáno jejich vzájemnou vzdáleností – jsou tak blízko sebe, že jim nezbyde nic, než si teplo předat.

Diamant – jeho perfektní krystalická mřížka neumožňuje žádný volný pohyb částic, tedy je izolantem. Ale právě díky dokonalosti jeho formy vede teplo – vibrace atomů se přes pravidelnou strukturu, kde jí nic nepřekáží, přenáší efektivně.

Zdroje

Tabulkové údaje mám z následujících tří zdrojů:

- *Table of specific heat capacities*. Online. Wikipedia. 2019. Dostupné z: https://en.wikipedia.org/wiki/Table_of_specific_heat_capacities#cite_note-hypph-3 [cit. 2023-11-25].
- *Solids, Liquids and Gasses – Thermal Conductivities*. Online. Engineeringtoolbox.com. 2003. Dostupné z: https://www.engineeringtoolbox.com/thermal-conductivity-d_429.html [cit. 2023-11-25].
- *Table of conductivity and resistivity at 20 °C of some materials*. Online. Faculty.ksu.edu.sa. 2022. Dostupné z: <https://faculty.ksu.edu.sa/sites/default/files/table.pdf> [cit. 2023-11-25].

Jako další jsem si dohledávala pár informací zde:

- *Krystalické a amorfní látky*. Online. Realisticky.cz. 2018. Dostupné z: <https://tinyurl.com/4aa79bd6> [cit. 2023-11-25].

Poslední odstavec řešení od Lucie vyžaduje dovysvětlení. To, co drží elektrony diamantu na místě, není krystalická mřížka, ale kovalentní vazby. Valenční elektrony v diamantu se účastní kovalentní vazby mezi konkrétními dvěma atomy, takže nepřeskakují mezi atomy tak snadno jako v kovové mřížce mezi jádry atomů kovu. Kovová vazba je jiný typ vazby, ve které elektrony volně přeskakují mezi jádry atomů kovu v mřížce. Toto přeskakování je důvod, proč jsou kovy dobrými tepelnými vodiči, a nižší pohyblivost elektronů v prostoru ve vazebné struktuře diamantu je důvod, proč je hodnota měrné tepelné kapacity diamantu nižší než pro kovy.

Směrovost chemických reakcí, část 1: entalpie (neplést s entropií)

První termodynamický zákon nám říká, že všechny formy energie jsou rovnocenné, tedy že se mezi sebou mohou přeměňovat. Nyní se podíváme na to, kdy se mezi sebou mohou formy energie (teplo, práce, jiné typy energie) přeměňovat. Děje mohou probíhat samovolně a nesamovolně.

U spousty dějů uhodnete, kterým směrem se stanou, čili který ze dvou stavů je počáteční a který je konečný. U hrnku s horkým čajem poznám, že teplo ze systému obsahujícího hrnek a čaj odchází, poznám, že stlačená pružinka se po uvolnění sama roztáhne, a takových příkladů najdeme spoustu. Ale co třeba taková chemická reakce? V hodinách chemie jste mohli slyšet, že chemická reakce je posunutá ve směru výchozích látek nebo produktů¹. Existuje kritérium, které nám umožní poznat, zda budou nějaké chemikálie mezi sebou reagovat (jsou v počátečním stavu před dějem a zreagují) nebo nebudou (existence dvou látek je konečný stav s nižší hodnotou celkové energie systému). Toto kritérium je Gibbsova volná energie. Značí se G a její jednotka je joule jako pro každou jinou formu energie. Definiční vzorec je $\Delta G = \Delta H - T\Delta S$. Pokud chcete vědět, jak se dobereme k definičnímu vzorci Gibbsovy volné energie, podívejte se na toto video². My se k tomuto vzorci dostaneme v příštím díle. Dnes si představíme veličiny, ze kterých se skládá a co tyto veličiny znamenají.

Nyní si můžeme představit definice samovolného a nesamovolného děje z hlediska termodynamiky, když už jsme si představili aktéry, kteří vystupují v jejich definicích. Postupně si je představíme blíže.

Samovolný termodynamický děj je ten, kde ve směru děje hodnota entropie narůstá a rozdíl hodnot Gibbsovy volné energie mezi počátečním a konečným stavem má zápornou hodnotu.

Nesamovolný termodynamický děj je ten, kde ve směru děje hodnota entropie klesá a rozdíl hodnot Gibbsovy volné energie mezi počátečním a konečným stavem má kladnou hodnotu.

Ve vzorci pro Gibbsovu energii jsou dvě nové fyzikální veličiny, tak si je pojďme představit v tomto díle. Je toho k představování spousta, takže se Gibbsově energii budeme věnovat i další díl.

Entalpie H není entropie S

Rozdíl hodnoty entalpie (ne entropie) mezi počátečním a konečným stavem značíme ΔH . Entalpie svou velikostí představuje energii uloženou v daném systému, tudíž její jednotkou jsou jouly. Pokud chcete vědět víc, toto video³ entalpii vysvětluje velmi pěkně a tento text jím je inspirován.

¹Posunutí chemické reakce ve směru produktů je termín, který říká, že reaktanty (tedy výchozí látky) chtějí reagovat, posun ve směru reaktantů říká, že výchozí látky reagovat nebudou, protože systém má menší Gibbsovu energii (její definice viz dále v textu) v případě, že se v něm nachází převážně výchozí látky.

²<https://www.youtube.com/watch?v=WTtx1aeC9PY>

³<https://www.youtube.com/watch?v=SV7U4yAXL5I>

Definiční vzoreček velikosti entalpie v jednom stavu je $H = U + pV$. S vnitřní energií U jsme se již potkali, jde o hodnotu energie, kterou má systém na svém energetickém účtu. Člen pV odpovídá součinu tlaku a objemu v daném stavu a reprezentuje energii, kterou systém stojí zabírat nějaký objem a vnitřní tlak. Nás často zajímá rozdíl hodnot entalpie mezi počátečním a konečným stavem, protože tato energie je právě ta, která se přeměňuje na jiné formy a dělá se systémem zajímavé věci. Rozdíl hodnot entalpie mezi dvěma stavy je tedy $\Delta H = \Delta U + p\Delta V$. V diferenciální formě lze tento vzoreček napsat takto: $dH = dU + pdV$. Na zemi je možné považovat tlak za konstantní (atmosférický tlak se mění velmi málo v porovnání s ostatními členy rovnice).

To je sice hezký vztah, ale k čemu nám jako chemikům bude? Na zodpovězení této otázky provedeme úpravu definičního vztahu entalpie. Použijeme vztahy pro nám již známou termodynamickou veličinu z předchozích dílů tématka, vnitřní energii: $\Delta U = Q + W = -p\Delta V + Q$. Tento vztah předpokládá, že při přechodu mezi dvěma stavy je tlak p konstantní. Tento předpoklad si můžeme u procesů (chemických reakcí) vystavených atmosférickému tlaku dovolit. Dosazení výrazu pro vnitřní energii do vzorce pro entalpii nám dá tento vztah: $\Delta H = \Delta U + p\Delta V = -p\Delta V + Q + p\Delta V = Q$. Takže hodnota rozdílu entalpie mezi dvěma stavy se rovná systémem přijatému nebo odevzdanému teplu během tohoto přechodu mezi dvěma stavy. Nezapomínejme, že tu počítáme s konstantním tlakem v systému, jinak by se nám dva členy tlaku a objemu neodečetly. Také si vzpomeňte na jeden příklad minulého dílu, kde se teplo počítalo jako měrná tepelná kapacita za standardního tlaku.

Úloha 3.1 [2b]: *Jak by vztah pro entalpii vypadal v diferenciální formě pro proměnný tlak systému během přechodu z počátečního stavu do konečného? Odvoďte jej a popište, co znamenají jednotlivé veličiny v odvozeném vztahu.*



Výpočty hodnot entalpie

Dobrá, takže když spočteme rozdíl entalpie mezi dvěma stavy, tak jsme o krok blíže k určení směru (nejen) chemického děje. Pro chemiky konkrétně to znamená, že počítáme rozdíl hodnot entalpie výchozích látek a produktů, tak to poté můžeme spolu s vypočteným rozdílem entropie (ten si ukážeme v příštím díle) použít, abychom zjistili, zda nám chemikálie za nějakých podmínek zreagují nebo ne.

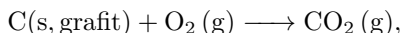
Na to, abychom určili tuto číselnou hodnotu, potřebujeme zavést něco, čemu se říká ve fyzice i chemii standardní stav. Proč chceme standardní stav? Neumíme najít, kolik je absolutní hodnota entalpie pro nějakou látku za teploty 0 K. Nelze jí prakticky dosáhnout a tudíž se za 0 K nedá měřit, takže hodnotu entalpie látky/molekuly na 0 K nemůžeme vzít jako nějaký referenční bod. Chemici si tedy nadefinovali takový stav, za kterého se dá provést většina experimentů, a to teplotu 298,15 kelvinů a tlak 101 235 Pa. Hodnoty entalpie pro molekuly jsou tedy vztaženy ke standardnímu stavu, kde je hodnota entalpie dle dohody brána jako nulová.



Standardní stav je stav (stejně jako počáteční nebo konečný stav), který má přesně specifikované hodnoty termodynamických veličin: teplotu 298,15 kelvinů a tlak 101 235 Pa.

Vzhledem k tomu, že nás dost často zajímají rozdíly hodnot entalpie mezi dvěma stavy, vlivy posunutí nulové hodnoty entalpie pro libovolnou chemickou sloučeninu do standardního stavu se často odečtou.

Nyní k praktické části výpočtů entalpie: Konkrétní příklad toho, jak může vypadat zadaná hodnota:



$$\Delta_r H^\circ = \Delta_f H^\circ = -393,5 \text{ kJ} \cdot \text{mol}^{-1}, T = 298,15 \text{ K}$$

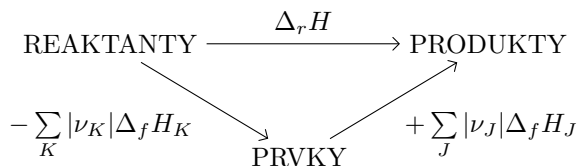
Indexy i značky rozdílů molární entalpie ΔH reflektují různá přídavná jména, která k entalpii můžeme přidat. Kolečko nahore odpovídá přídavnému jménu standardní. Další možnosti mohou vypadat například takto:

- reakční (ta se vztahuje k rozdílu hodnot entalpií pro stav před a po reakci)
- slučovací entalpie (často značeno pomocí spodního indexu *sl*): entalpie potřebná na vznik molekuly z prvků
- často najdete i entalpii potřebnou pro přechod mezi dvěma formami jednoho prvku (grafit–diamant pro uhlík, konformace ledu apod.)
- skupenská (kondenzační, výparné, entalpie tuhnutí): teplo musíte dodat nebo odebrat i při skupenské přeměně (hlídejte si, zde se vám tabelovaná entalpie, kterou najdete, váže i ke správnému skupenství, které máte v chemické reakci).

Hodnoty reakčních entalpií jsou dost často tabelované právě proto, že fyzici a chemici nechali danou reakci proběhnout v kalorimetru a změřili velikost uvolněného tepla.

Někoho z vás možná napadlo, zda nepotřebujeme vědět, kudy chemická reakce běží cestou z výchozího stavu do stavu konečného. Není to potřeba, protože entalpie je definovaná jako stavová funkce. Váže se tedy ke stavu, nikoliv k procesu. Proto si můžeme vybrat libovolný počet mezikroků, přes který naši reakci provedeme z reaktantů na produkty. Často se ubíráme cestou, kdy výchozí látky rozložíme na produkty a z produktů uděláme reaktanty (viz obr. 1). Jde to však i přes jiné chemické reakce. Podívejte se na to, že zde vystupují i stechiometrické koeficienty reakce ν_i a všimněte si i znamének před sumami. Tabelované hodnoty entalpie jako reakční entalpie pro spálení grafitu kyslíkem jsou pro děj ve směru reakce, hodnota molární reakční entalpie reakce opačného směru by měla opačné znaménko.

Když hledáte hodnoty standardní entalpie, podívejte se, jakými hodnotami tlaku a teploty je definován standardní stav.



Obrázek 1: Princip výpočtu entalpie chemické reakce (od stavu REAKTANTY do stavu PRODUKTY) přes jiný stav (stav PRVKY). Spodní index f značí slučovací teplo nějaké molekuly z prvků, veličiny značené K odpovídají reaktantům, J produktům.

Co když ale proces probíhá za jiné teploty, než za které je naše hodnota rozdílů molární entalpie uvedená? Vzpomeňte si, že pro konstantní tlak je rozdíl entalpie roven vyměněnému teplu (vysvětlení je ve vzorovém řešení příkladu úlohy 2.2 publikovaném v tomto čísle M&M). Můžeme tu tedy uplatnit něco, čemu se říká Kirchhoffův zákon:

$$\Delta_r H_m(T_2) = \Delta_r H_m(T_1) + \int_{T_1}^{T_2} \Delta_r C_{p,m} dT$$

Vychází z definice molární tepelné kapacity $c_p = C_p/n$ za konstantního tlaku. Pro jednodušší výpočty předpokládáme, že C_p je nezávislé na teplotě. Tento předpoklad je dostatečně podobný realitě u většiny látek na rozumném rozsahu teplot, to nám ostatně ukázala úloha z minulého čísla.

Veličiny $\Delta_r H_m(T_1)$ a $\Delta_r H_m(T_2)$ jsou molární entalpie procesu mezi dvěma stavy 1 a 2 o teplotách T_1 a T_2 , tabelovanou nebo spočtenou máte jen jednu z entalpií. Veličina $\Delta_r C_{p,m}$ je rozdíl molárních tepelných kapacit koncových produktů a výchozích látek dle vzorce

$$\Delta C_{p,m} = \sum_{\text{kon}} (C_{p,m})_{\text{kon}} - \sum_{\text{vých}} (C_{p,m})_{\text{vých}}$$

Nyní máte spoustu informací na to, abyste si výpočty entalpie vyzkoušeli.

Úloha 3.2 [5,5b a více]: *Reakce $C_2H_4O(g) + H_2O(l) \longrightarrow C_2H_6O_2(l)$ probíhá při $25^\circ C$ a tlaku $101,325 \text{ kPa}$. Výpočet proveďte pro případ, že všechny látky jsou v plynné (platí stavová rovnice ideálního plynu) a nebo v kapalně fázi tak, jak je uvedeno v reakci. Data při podmínkách reakce:*

	C_2H_4O	H_2O	$C_2H_6O_2$
$\Delta H_{sl}^\circ(298,15 \text{ K}, (g))/(\text{kJ/mol})$	-52,60	-241,827	-389,32
$\Delta H_{vyp}^\circ(298,15 \text{ K})/(\text{kJ/mol})$	26,20	44,0	71,20
$\rho/(\text{g/cm}^3)$	0,8610	0,9971	1,1094

1. Na základě následujících dat vypočítejte standardní změnu entalpie při hydrataci ethylenoxidu na glykol.



2. Vypočtete změnu molárního objemu během reakce.

3. Vypočtete změnu vnitřní energie reakce.

Pro získání extra bodů popište, jaké předpoklady platí pro vzorce, co jste použili (zda plyn musí být ideální, zda něco zanedbáváte apod.). Zamyslete se nad tím, kdy vámi vymyšlený vzorec použít nelze.



Úloha 3.3 [5b]: Při experimentu, jehož cílem bylo určení slučovací entalpie celulózy, bylo zjištěno, že během dokonalého spálení $m = 0,6$ g polysacharidu celulózy (chemický vzorec $[C_6H_{10}O_5]_{12}$) v kalorimetrické bombě přešlo při $25^\circ C$ do okolí teplo $Q = 2000$ J. Pomocí slučovacích tepel $CO_2(g)$ a $H_2O(l)$ určete slučovací teplo tohoto polysacharidu. Potřebujete molární hmotnost celulózy M a slučovací tepla při $25^\circ C$, která naleznete v tabulkách nebo na internetu. Reakce probíhá za stálého objemu. Je reakce endotermická nebo exotermická? Proč?

Nápověda: napište si rovnici, objemy látek v kapalném a tuhém stavu zanedbáváme vůči velikosti objemu plynných látek. Člen pV se vyskytuje i v rovnici pro ideální plyn. Při konstantním objemu systému je uvolněné teplo Q během děje rovno změně molární vnitřní energie ΔU_{sp}° podle rovnice:

$$\Delta U_{sp}^\circ = \frac{QM}{m}$$

Pája, Helča a Pavel; termodynamika@ledoian.cz
odevzdávejte do odevzdávátka



Li-bub (Limonádovo-bublínková baterie)

7b

Dr.^{MM} Martin Těšitel

Poznámka redakce: Spousta organizátorů Martina chválí za to, jak hezkou a přehlednou strukturu článek má. Hezky je vidět, o čem v danou chvíli uvažuje, vyjadřuje se jasně a k věci. Článek má uvedené zdroje, ze kterých Martin čerpal, tam se také můžete podívat, až budete na jeho článek reagovat. Proto je mimo jiné dobré ke svým textům uvádět použité zdroje, pokud nějaké jsou.

Tento článek má však jisté mouchy, zejména při úvaze, že je zanedbáno, že CO_2 se rozpouští ve vodě a je ho tam tudíž více, než je obsaženo ve vzduchu.

Secke, kde se Martin zamýšlí nad praktickou realizací, obsahuje myšlenky, na které můžete navázat. Tady je místo pro vás, ostatní řešitele. Zareagujte jiným článkem na tento Martinův. Jak byste Li-bub baterii v realitě postavili? Myslíte si, že by byla tak účinná, jak Martin spočítal? Proč ano/ne?

Úvod

Otevřeli jste už někdy perlivou limonádu a říkali si, jestli by nemohla udělat něco víc, než jen cvaknout a zasyčet? Že by mohla třeba pískat, hrát vaši oblíbenou písničku, nebo blikat?

V tomto krátkém (a doufám, že zajímavém, nebo alespoň vtípném) článku se budu zabývat využitím energie natlakovaného CO_2 ke svícení LED diodou připevněnou na víčku.

Kolik energie limonáda skrývá?

Jako modelovou limonádu berme 1,5l láhev Coca-Coly o obsahu CO_2 8 g/l a pokojové teplotě⁴. Z grafu uvedeného v dokumentu o kvasném průmyslu⁵ jsem vyčetl, že v takové láhvi bude mít CO_2 tlak cca 5 atm.

Měřením⁶ jsem zjistil, že v láhvi je nad hladinou nápoje cca 130 ml CO_2 , což mi spolu s poznatkem o hodnotě tlaku umožňuje vypočítat⁷, že se při otevření láhve uvolní cca 100 J.

$$p_1 = 5 \text{ atm} = 5 \cdot 10^5 \text{ Pa}$$

$$p_2 = 1 \text{ atm} = 10^5 \text{ Pa}$$

$$V = 1,32 \cdot 10^{-4} \text{ m}^3$$

⁴<https://cs.wikipedia.org/wiki/Sodovka>

⁵<https://kvasnyprumysl.cz/pdfs/kpr/1962/04/05.pdf> (obr. 1 str. 2)

⁶U původní nenačaté láhve Coca-Coly jsem si vyznačil výšku hladiny. Po vypití obsahu jsem do láhve nalil vodu do výšky původní hladiny a láhev zvažil. Poté jsem ji dolil vodou až po okraj a znovu ji zvažil. Počítáme-li s tím, že voda má hustotu přesně 1000 kg/m³, je rozdíl hmotnosti v gramech roven objemu vzduchu nad hladinou limonády v mililitrech.

⁷Vzoreček pro výpočet práce jsem našel v této bakalářské práci na stránce 16 https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=27235



$$W = p_1 \cdot V \cdot \ln\left(\frac{p_1}{p_2}\right) = 5 \cdot 10^5 \cdot 1,3 \cdot 10^{-4} \cdot \ln\left(\frac{5 \cdot 10^5}{10^5}\right) \text{ J} = 105 \text{ J}$$

Předpokládejme, že účinnost našeho generátoru⁸ bude 15 %, a výsledné množství získané energie tedy bude 16 J. Pro srovnání, maximální kapacita knoflíkové baterie SR65 je⁹ 86 J.

Dejme tomu, že budu chtít svítit pouze jednou standardní LED diodou s výkonem 0,015 W o svítivosti¹⁰ 1,3 cd.

Taková dioda by byla s touto energií schopna svítit 1060 sekund, což je zhruba čtvrt hodiny.

Jak přeměnit energii limonády na elektrickou energii?

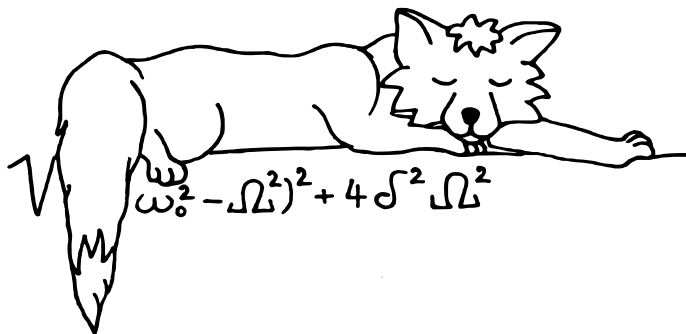
Když jsem uvažoval nad tímto problémem, napadlo mě několik způsobů přeměny rozdílu tlaku na elektrickou energii. Ty použitelné nebo zajímavé, o kterých zde budu psát, pracují vždy na principu dynama, nebo kmitajícího plíšku.

Mým prvním nápadem byla malá turbína spojená s dynamem, což je vlastně princip, na kterém funguje většina současných elektráren.

Druhý způsob spočívá v rozkmitání plíšku s magnetem v blízkosti cívky, nebo v rozkmitání plíšku s piezoelektrickým krystalem. Oba tyto způsoby by měly mít přibližně stejnou účinnost, která bude pravděpodobně řádově podobná účinnosti turbíny s dynamem.

Závěr

Jak se zdá, je svícení LED diodou na víčku proveditelné, ale pravděpodobně by bylo levnější použít místo generátoru obyčejnou malou baterii.



⁸Jedná se pouze o odhad

⁹https://cs.wikipedia.org/wiki/Knofl%C3%ADkov%C3%A1_baterie

¹⁰<https://www.tme.eu/Document/d32745b75c434cebe6f86eeb0cb5ee80/OSXX060341F.pdf>
(pro zelenou barvu)

Téma 2 – Programování a dokazování v Leanu

Díl 3: Elementární algebraické manipulace

Lean je nejen programovací jazyk, ale také dokazovací jazyk. Píší se v něm matematické důkazy, jejichž správnost je okamžitě zkontrolována počítačem. To je v současné době hlavní využití Leanu. Zatímco dobrých programovacích jazyků existuje spousta, v oblasti „interactive theorem proving“ Lean hraje prim.

Rovnosti

Řekněme, že chceme ověřit, že dvojnásobek každého přirozeného čísla je roven součtu tohoto čísla se sebou:

```
theorem dva_krat (n : ℕ) : 2 * n = n + n := sorry
```

Klíčové slovo `theorem` uvozuje začátek tvrzení. Jméno tvrzení `dva_krat` slouží k tomu, abychom se na něj mohli později odkazovat. Argument `(n : ℕ)` říká, že tvrzení má platit pro jakékoliv přirozené číslo; napsáním `n` se na něj budeme později odkazovat. Znění tvrzení začíná hned za dvojtečkou mimo závorku. Symbol `:=` odděluje tvrzení a jeho důkaz. Klíčové slovo `sorry` říká, že tvrzení zatím není dokázáno.

Díky `sorry` můžeme zkompileovat rozpracovaný kód bez syntaktické chyby. Po většinu vaší práce v Leanu bude `sorry` ve vašem kódu na minimálně jednom místě. Po zmazení posledního `sorry` je vaše práce hotová. Klíčové slovo `sorry` bývá ve vývojových prostředích zvýrazněno jasně červenou barvou, aby člověk nepřehlédnul, že důkaz ještě není hotový.

Nyní, když používáme knihovnu `Mathlib`, budeme psát symbol `ℕ` (vložen pomocí `\N` ve VS Code) jako alias pro typ `Nat` (tj. přirozená čísla) a symbol `ℤ` (vložen pomocí `\Z` ve VS Code) jako alias pro typ `Int` (tj. celá čísla).

Jako první dává smysl podívat se, jestli toto tvrzení (nebo nějakou jeho obecnější verzi) už někdo nedokázal. K tomuto účelu slouží taktika `library_search` volaná takto (na moment teď ignorujeme, co znamená slovo „taktika“):

```
theorem dva_krat (n : ℕ) : 2 * n = n + n := by library_search
```

Lean oznámí `exact two_mul n` na pravé straně okna. Z toho plyne, že můžeme existující tvrzení `two_mul` použít jako důkaz našeho tvrzení. Napíšeme to takto:

```
theorem dva_krat (n : ℕ) : 2 * n = n + n := two_mul n
```

Protože máme naimportovanou správnou knihovnu, kompilátor si na nic nestěžuje. Dojde-li ke stížnostem na vašem počítači, podívejte se na plné verze našich zdrojových kódů¹¹ a zkopírujte patřičné importy navrch vašeho zdrojového kódu.

Úloha 3.1 [0,5b]: Pomocí `library_search` najděte důkaz tohoto tvrzení:

```
theorem na_druhou (a : ℚ) : a ^ 2 = a * a := sorry
```

¹¹<https://github.com/madvorak/lean-mam/blob/main/mam/Cislo3.lean>





Jak by to bylo s trojnásobkem čísla?

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := sorry
```

Když zkusíme `library_search` jako minule, nic užitečného nám to nenajde. Všechny řádky na pravé straně obrazovky začínají slovem `refine` namísto `exact` označujícího přesný nále. Nic z toho se nám hodit nebude.

Možným řešením je, jak už to v matice bývá, rozbít problém na kousky. Nabízí se výraz $2 * n + n$ jako mezistupeň mezi levou a pravou stranou. Pozorně si přečtete následující syntaxi:

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := by
  convert_to 3 * n = 2 * n + n
  sorry
  sorry
```

Jako první vás asi zarazilo, proč je tam `sorry` dvakrát. To je proto, že taktika `convert_to` vygenerovala dva cíle. První cíl $n + n = 2 * n$ vyjadřuje, co je potřeba dokázat, aby došlo k převodu původního tvrzení ($3 * n = n + n + n$) na zvolené tvrzení ($3 * n = 2 * n + n$). Druhý cíl $3 * n = 2 * n + n$ říká, že to je potřeba dokázat. První `sorry` říká, že první cíl dokážeme později. Druhé `sorry` říká, že druhý cíl taktéž dokážeme později. Nyní si můžeme vybrat, na kterém podproblému chceme začít pracovat.

Když se podíváme na první cíl ($n + n = 2 * n$), zaradujeme se, protože to už přeci máme dokázané! Pouze si musíme dát pozor, že levá a pravá strana jsou obráceně. Proto zavoláme `dva_krat` takto:

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := by
  convert_to 3 * n = 2 * n + n
  exact symm (dva_krat n)
  sorry
```

Funkce `symm` prohazuje levou a pravou stranu rovnosti. Slovo `exact` si vysvětlíme za okamžik.

O vyřešení druhého cíle ($3 * n = 2 * n + n$) požádáme `library_search` takto:

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := by
  convert_to 3 * n = 2 * n + n
  exact symm (dva_krat n)
  library_search
```

Objeví se `exact Nat.succ_mul 2 n` na pravé straně okna. Zkopírujeme to na místo `library_search` a máme hotový důkaz:

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := by
  convert_to 3 * n = 2 * n + n
  exact symm (dva_krat n)
  exact Nat.succ_mul 2 n
```


Než postoupíme ke složitějším tvrzením, osahejme si vývojové prostředí. Když klikneme myší na konec prvního řádku (na klíčové slovo `by` nebo za něj), uvidíme na pravé straně okna toto:

```
n : ℕ
├ 3 * n = n + n + n
```

Symbol `├` označuje cíl (co ještě chybí). Všechny předchozí řádky (zde to je jen jeden řádek) říkají, co máme k dispozici. Když klikneme o řádek níže (tj. na konec druhého řádku), uvidíme toto:

```
n : ℕ
├ n + n = 2 * n
n : ℕ
├ 3 * n = 2 * n + n
```

Zde vidíme oba cíle, které byly otevřené po zavolání `convert_to` v našem důkazu. Zkuste si, co se na pravé straně okna objeví, když kliknete ještě níže. Vidíme, že v tzv. lokálním kontextu je teď pouze přirozené číslo `n` a nic dalšího, ale kromě čísel (obecně kromě hodnot) bychom tam mohli vidět i předpoklady nebo pomocná tvrzení, která jsme již dokázali uvnitř rozpracovaného důkazu. Lokální kontext zobrazuje všechno, co máme „po ruce“.

Technicky vzato, Lean nerozlišuje mezi daty a tvrzeními, mezi hodnotami a důkazy. Všechno to jsou tzv. termy. Možná se teď ptáte, co nejsou termy? Klíčová slova a s nimi spjaté symboly nejsou termy. Dále existují tzv. taktiky, což není to samé jako termy. Taktiky jsou nápodobou tzv. příkazů, což je pojem, který možná znáte ze světa imperativního programování. Potkaná slova `library_search` a `convert_to` jsou příklady taktik. Když chceme zavolat taktiku v místě, kde se očekává term, klíčovým slovem `by` se přepneme do tzv. `tactic` módu. Naopak slovo¹² `exact` nám umožňuje v `tactic` módu napsat term. Proto také `library_search` oznamoval `exact` následovaný tzv. `proof` termem v případě úspěchu. V místě, kde by `library_search` nabídl `exact two_mul n` jsme mohli přímočaře nahradit `library_search` za `exact two_mul n` a tím by vznikl tento důkaz:

```
theorem dva_krat (n : ℕ) : 2 * n = n + n := by exact two_mul n
```

Protože však víme, že `by` a `exact` dělají přesně opačnou věc, navzájem se jejich účinky vyruší, a proto uvedená dvojice `by exact` dělá to samé, jako kdyby tam nebylo ani jedno. Lepší je ten důkaz, který jsme použili my:

```
theorem dva_krat (n : ℕ) : 2 * n = n + n := two_mul n
```

Zde slouží `two_mul n` jako dotyčný `proof` term. Není potřeba otevírat `tactic` mód. Důkaz tvrzení `dva_krat` je tak krátký. V kontrastu s tím je možné poukázat,

¹²Technicky vzato `exact` je taktika. A aby to bylo ještě víc matoucí, `sorry` je taktika i term (zkuste si `exact sorry` v nějakém nedokončeném důkaze). Věřili jste nám, když jsme vám tvrdili, že `sorry` je klíčové slovo? Sorry!



že následující důkaz taktéž projde kompilátorem bez chyby, ale je ošklivý a matoucí:

```
theorem dva_krat (n : ℕ) : 2 * n = n + n :=
  by exact by exact by exact by exact two_mul n
```

Prosím, nepište důkazy takhle.

Možná jste si říkali, proč jsme museli tak dlouho bušit do klávesnice, abychom dokázali tak jednoduché tvrzení, jako bylo `tri_krat` dokazované před chvílí. Tato námitka by byla na místě. Opravdu není potřeba psát třířádkový důkaz. Pro tyto účely je v knihovně Mathlib předpřipravená taktika `ring` s úplně jednoduchým voláním:

```
theorem tri_krat (n : ℕ) : 3 * n = n + n + n := by ring
```

To je celý důkaz! Taktika `ring` umí automaticky provést rozmanité posloupnosti algebraických manipulací, které využívají vlastností součtů, rozdílů, součinů a mocnin (ale ne podílů).

Pokud se aktivně zajímáte o algebru, nejspíš jste někdy slyšeli slovo „okruh“ (anglicky „ring“); podle toho se ta taktika jmenuje. Ukažme si několik dalších tvrzení, která zvládne taktika `ring` dokázat úplně sama:

```
theorem soucet_na_druhou (x y : ℤ) :
  (x + y) ^ 2 = x^2 + 2*x*y + y^2 := by
  ring
```

```
theorem rozdil_na_treti (x y : ℚ) :
  (x - y) ^ 3 = x^3 - 3*x^2*y + 3*x*y^2 - y^3 := by
  ring
```

```
theorem rozdil_patych_mocnin (x y : ℝ) :
  x^5 - y^5 = (x - y) * (x^4 + x^3*y + x^2*y^2 + x*y^3 + y^4) := by
  ring
```

Psaní mezer kolem některých výskytů binárních operátorů a nepísání kolem jiných je čistě estetická záležitost. Většinou se snažím formátováním kódu naznačit, jaké jsou priority operátorů, aby se kód snáze četl. Nemá to vliv na jeho funkci.

Když si chceme něco jen vyzkoušet a nedávat tomu název, místo klíčového slova `theorem` to uvedíme klíčovým slovem `example` jako zde:

```
example (n : ℕ) : 2 ^ (n+3) = 8 * 2^n := by ring
```

Správnost důkazu je v případě `example` kontrolována stejně, jako kdyby to byl `theorem`. Zkuste si nějak poškodit tohle tvrzení (aby už neplatilo) a spatříte, že to kompilátor označí za chybu.

Následující příklad ukazuje, jak se pracuje s předpoklady (jméno taktiky `rw` pochází z anglického „rewrite“):

```
example (n : ℕ) (n_je_pet : n = 5) : n - 1 = 4 := by
  rw [n_je_pet]
```

Taktika `rw` nahradí všechny výskyty levého výrazu v našem cíli za pravý výraz. Kdyby byl předpoklad zapsaný zrcadlově, mohli bychom použít funkci `symm` podobně, jako jsme to už jednou udělali:

```
example (n : ℕ) (n_je_pet : 5 = n) : n - 1 = 4 := by
  rw [symm n_je_pet]
```

Tohle však není idiomatické. V případě taktiky `rw` je zvykem obrácené přepsání (tj. namísto substituce levé strany za pravou děláme substituci pravé strany za levou) značit symbolem šipka doleva:

```
example (n : ℕ) (n_je_pet : 5 = n) : n - 1 = 4 := by
  rw [←n_je_pet]
```

V rámci jedné hranaté závorky je možné provést i více substitucí:

```
example (a b c : ℝ)
  (a_je_dva : a = 2) (b_je_tri : b = 3) (c_je_pet : c = 5) :
  a + b = c := by
  rw [a_je_dva, b_je_tri, c_je_pet]
  ring
```

Skvělé je, že můžeme uvnitř vývojového prostředí klikat i na jednotlivá místa uvnitř hranaté závorky a vidět tam, jak se cíl postupně mění. Po provedení třetí substituce je cíl $2 + 3 = 5$ a Lean ho neuzavřel automaticky. Proto použijeme taktiku `ring` k dokončení důkazu.

Pomocí `at` můžeme změnit chování, že se přepis provede v jiném předpokladu (namísto v cíli). Tomuto příkladu asi nebudete rozumět zcela, ale použití taktiky `rw by` mělo být jasné (získáme $2 = 3$ z předpokladů):

```
example (a : ℝ) (a_je_dva : a = 2) (a_je_tri : a = 3) : False := by
  rw [a_je_dva] at a_je_tri
  norm_num at a_je_tri
```

Teď se podívejme na příklad zahrnující dělení:

```
example (x : ℝ) (xnn : x ≠ 0) : x^2 / x = x := by
  field_simp
  ring
```

Taktika `field_simp` zjednoduší cíl na $x^2 = x * x$ (což poté `ring` umí uzavřít). Stejným způsobem můžeme dokázat následující tvrzení:

```
theorem plus_preveracena (x : ℝ) (xnn : x ≠ 0) :
  x + 1/x = (x^2 + 1) / x := by
  field_simp
  ring
```



Tento příklad asi teď nikoho nepřekvapí:

```
example (x y z : ℝ) (xnn : x ≠ 0) :
  x*y*z + 3*y*z*x - 2*z*x*y = y*x*z + x^2*z*y/x := by
  field_simp
  ring
```

Nerovnosti

```
example (x y z : ℝ) (xy : x ≤ y) (yz : y ≤ z) : x ≤ z :=
  sorry
```

Požádáním `library_search` dostaneme nějakou variantu tohoto:

```
example (x y z : ℝ) (xy : x ≤ y) (yz : y ≤ z) : x ≤ z :=
  xy.trans yz
```

Pro složitější zadání nám však už `library_search` nedá dobrou odpověď (řádky začínající slovem `refine` ignorujeme):

```
example (a b c d : ℝ)
  (abcd : a + b + c ≤ 2 * d) (ab : a ≤ b) (ac : 2 * a ≤ c) :
  2 * a ≤ d :=
  sorry
```

Inu, nemůžeme čekat, že by v knihovně známých matematických výsledků bylo napsáno zrovna takovéhle tvrzení. Naštěstí však existuje taktika `linarith` pro řešení soustav lineárních nerovnic. Tento kód úspěšně projde kompilátorem:

```
example (a b c d : ℝ)
  (abcd : a + b + c ≤ 2 * d) (ab : a ≤ b) (ac : 2 * a ≤ c) :
  2 * a ≤ d := by
  linarith
```

Tohle však už neprojde:

```
example (x y : ℝ) (xy : x ≤ y) : x ≤ y + y*y := by linarith
```

Z principu, že `linarith` řeší jen lineární nerovnice, mu uniká, že `y*y` je nezáporné číslo. Taktika `nlinarith` však o těchto základních nelineárních vztazích ví. Proto tenhle kód už projde:

```
example (x y : ℝ) (xy : x ≤ y) : x ≤ y + y*y := by nlinarith
```

Také následující příklad `nlinarith` vyřeší (kde by `linarith` selhala):

```
example (x y : ℝ) (x_zaporne : x < 0) (y_zaporne : y < 0) :
  x * 7 * y > 0 := by
  nlinarith
```

Poradí si `nlinarith` i tady?

```
example (x y : ℝ) : x*x - 2*x*y + y*y ≥ 0 := by nlinarith
```

Tenhle důkaz už neprojde. Jak bychom to tedy dokázali? Matematiky-znalý čtenář nahlédne, že výraz $x*x - 2*x*y + y*y$ je roven $(x - y)^2$ a ten je samozřejmě nezáporný. Napovmězme tedy počítači, že takhle ten důkaz sestaví!

```
example (x y : ℝ) : x*x - 2*x*y + y*y ≥ 0 := by
  convert_to (x - y) ^ 2 ≥ 0
  ring
  nlinarith
```

Teď už `nlinarith` uspěl. Opět čtenáři doporučujeme proklikat se důkazem. Následuje složitější příklad stejného typu:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 81 ≥ 0 :=
  sorry
```

Opět využijeme tzv. převod na čtverec a důkaz je hotový:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 81 ≥ 0 := by
  convert_to ((2*x - 3) ^ 2) ^ 2 ≥ 0
  ring
  nlinarith
```

S nalezením správného rozkladu nám Lean nepomůže; to je práce pro člověka (nemusíme se ale bát, že bychom v rozkladu udělali chybu a důkaz by byl neplatný; Lean nás nenechá dokázat nic nepravdivého). Naopak nám Lean může práci zkomplikovat. Uvažme skoro stejný příklad:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 100 ≥ 0 :=
  sorry
```

Po vzoru minulého důkazu bychom mohli chtít začít tento důkaz takhle:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 100 ≥ 0 := by
  convert_to ((2*x - 3) ^ 2) ^ 2 + 19 ≥ 0
  sorry
  sorry
  sorry
  sorry
```

Náš záměr je převést zadaný výraz na součet dvou výrazů, které jsou oba na první pohled nezáporné. Lean bohužel provede něco velmi ošklivého! Podívejte se, jaké cíle se otevřely (kvůli nim jsme taky museli napsat `sorry` čtyřikrát):

```
⊢ 16 * x^4 - 96 * x^3 + 216 * x^2 - 216 * x = ((2 * x - 3) ^ 2) ^ 2
⊢ 100 = 19
⊢ 100 = 19
⊢ ((2 * x - 3) ^ 2) ^ 2 + 19 ≥ 0
```

Takže tudy cesta nevede... Mohli bychom si pomoci tím, že bychom upravili „hloubku konverze“ takto:



```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 100 ≥ 0 := by
  convert_to ((2*x - 3) ^ 2) ^ 2 + 19 ≥ 0 using 1
  ring
  nlinarith
```

Zde to zabralo, ale poněvadž taktika `convert_to` není samospásná, ukážeme si jiný, spolehlivější přístup – taktikou `have` si vytvoříme lokální pomocné tvrzení:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 100 ≥ 0 := by
  have pomocne : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 81 ≥ 0
  • sorry
  sorry
```

Nyní jsou otevřené dva cíle. První z nich je dokázat to pomocné tvrzení. Druhý cíl je dokázat původní tvrzení. K čemu jsme si pomohli? V lokálním kontextu pro druhý cíl se nachází naše pomocné tvrzení. Můžeme ho tedy použít k důkazu původního tvrzení. Tohle je hotový důkaz:

```
example (x : ℝ) : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 100 ≥ 0 := by
  have pomocne : 16*x^4 - 96*x^3 + 216*x^2 - 216*x + 81 ≥ 0
  • convert_to ((2*x - 3) ^ 2) ^ 2 ≥ 0
  • ring
  nlinarith
  linarith
```

Všimněte si puntíku na začátcích některých řádků. Tyto puntíky napíšeme pomocí zpětného lomítka a tečky. Ty jsou nepovinné, ale pomáhají nám v orientaci. Společně s nimi také používáme dvě mezery. O tyto znaky se však zajímá i kompilátor. Zkuste si smazat ty „nadbytečné“ mezery i puntíky a podívejte se, jak se změní informace zobrazované na pravé straně okna.

Na závěr si ukážeme jeden náročnější důkaz nerovnosti. Chceme dokázat, že pro každé kladné x platí $x + \frac{1}{x} \geq 2$. To je dostatečně náročné tvrzení na to, abychom neměli naději, že ho Lean zvládne dokázat automaticky. Zamysleme se nad nějakou sérií nerovností, počínaje něčím triviálním, která by ke kýžené nerovnosti vedla. Já jsem přišel s následujícím plánem důkazu:

$$\begin{aligned}
 (x - 1)^2 &\geq 0 \\
 x^2 + 1 - 2x &\geq 0 \\
 x^2 + 1 &\geq 2x \\
 \frac{x^2 + 1}{x} &\geq \frac{2x}{x} \\
 \frac{x^2}{x} + \frac{1}{x} &\geq \frac{2x}{x} \\
 x + \frac{1}{x} &\geq 2
 \end{aligned}$$

Z konceptuálního hlediska se jedná o tzv. dopředný důkaz. V Leanu to bude série taktik `have` s krátkými důkazy pro každé tvrzení. Vzhledem k tomu, že každé tvrzení přímo vyplývá z předchozího, nemusíme ta tvrzení pojmenovávat. Místo toho necháme Lean, aby vygeneroval automatický identifikátor `this` pro každé dílčí tvrzení, přičemž bude vždy možné odkazovat se pomocí `this` na nejnovější dokázané tvrzení (obecně by se jednalo o poslední nepojmenované tvrzení). Toto je hotový důkaz:

```
example (x : ℝ) (hx : x > 0) : x + 1/x ≥ 2 := by
  have : (x - 1) ^ 2 ≥ 0
  • exact pow_two_nonneg (x - 1)
  have : x*x + 1 - 2*x ≥ 0
  • convert this
    ring
  have : x*x + 1 ≥ 2*x
  • exact le_of_sub_nonneg this
  have : (x*x + 1) / x ≥ 2*x / x
  • have levy_citatel_nezap : x*x + 1 ≥ 0
    • nlinarith
    have samozrejmost : x ≤ x
    • exact refl x
    exact div_le_div levy_citatel_nezap this hx samozrejmost
  have : x*x/x + 1/x ≥ 2*x/x
  • convert this
    exact div_add_div_same (x*x) 1 x
  convert this
  • simp
  • convert_to 2 = 2 * (x / x)
    • exact mul_assoc 2 x x-1
  convert_to (2 : ℝ) = 2 * 1
  • have : x ≠ 0
    • exact ne_of_gt hx
    exact div_self this
  ring
```

Zatímco `convert_to` nastavuje nový cíl, na který nejprve převedeme současný cíl a poté ho dokážeme, taktika `convert` převádí aktuální cíl na již dokázané tvrzení (toto je zjednodušený pohled). Pochopitelně by nám stačilo mít jen jednu z nich; mezi `convert` a `convert_to` lze přecházet pomocí `exact` respektive `have` (se změnou pořadí věcí v důkaze).

Všimněte si, že na řádku `convert_to (2 : ℝ) = 2 * 1` jsme museli explicitně uvést, že dvojka je reálné číslo, protože `2 = 2 * 1` by Lean považoval za tvrzení o přirozených číslech.



Ve všech následujících úlohách je úkolem dokázat zadané tvrzení. Plným počtem bodů bude odměněno řešení, které

1. se zkompiluje bez chyby;
2. zachovává zadané znění tvrzení (zkopírujte si zadání¹³ z repozitáře);
3. neobsahuje **sorry** ani jeho ekvivalent.

Pokud vám nepůjde dokázat „celé“ tvrzení, můžete porušit třetí instrukci (tj. nechat nějaké neuzavřené cíle jako **sorry** ve vašem řešení). Pokud bude váš důkaz „na správné cestě“, odměníme vás částečným počtem bodů. Řešení porušující první instrukci nebo druhou instrukci dostane nula bodů.

Pokud odevzdáváte neúplné řešení (důkaz obsahující **sorry**), zdokumentujte svoje řešení (co zbývá dokázat a proč si myslíte, že to platí). Pokud odevzdáváte úplné řešení, nedokumentujte ho (odevzdejte zdrojový kód bez komentářů).

Úloha 3.2 [1b]:

```
example (a b c d : ℕ)
  (a_je : a = b + d) (b_je : b = a * a)
  (c_je : c = b + d) (d_je : d = c * c) :
  b ^ d = d ^ b :=
sorry
```

Úloha 3.3 [1b]:

```
example (a b c d : ℤ)
  (a_je : a = d ^ 4) (b_je : b = 1 / c)
  (c_je : c = a - b) (d_je : d = 4 * a) :
  (a + b) ^ 2 - c ^ 2 = b * d :=
sorry
```

Úloha 3.4 [2b]:

```
example (x : ℝ) :
  50*x^2 - 126*x + 96 ≥ 0 :=
sorry
```

Úloha 3.5 [2b]:

```
example (x y : ℝ) :
  2 * x^3 * y^3 ≤ x^4 * y^2 + x^2 * y^4 :=
sorry
```

¹³<https://github.com/madvorak/lean-mam/blob/main/mam/Zadani3.lean>

Úloha 3.6 [2b]:

```
example (x y z : ℝ) :
```

```
  4*x^2 + 12*x*y - 4*x*z + 9*y^2 - 6*y*z + z^2 ≥ 0 :=  
  sorry
```

**Úloha 3.7** [9b]:

```
example (a b : ℝ) (ha : 0 < a) (hb : 0 < b) :
```

```
  1 / a + 1 / b ≤ a / b^2 + b / a^2 :=  
  sorry
```





Řešení úloh z 2. dílu

Zadání:

Napište funkci generující seznam (zadané délky) čísel, které jsou celočíselnými násobky sedmi.

Řešení:

```
def nasobky_sedmi : Nat → List Nat
| 0 => [ ]
| n+1 => (7 * n) :: (nasobky_sedmi n)
```

Zadání:

Objasněte, proč je následující implementace určení, zda je seznam konstantní, chybná:

```
def je_konst {T : Type} [DecidableEq T] : List T → Bool
| [ ] => true
| [ _ ] => true
| prvni :: druhy :: zbytek => (prvni = druhy) && je_konst zbytek
```

Řešení:

Zkuste si `#eval je_konst [1, 1, 2, 2]` a hned uvidíte, co je špatně.

Zadání:

Naprogramujte součin seznamu celých čísel.

Řešení:

```
def soucin : List Int → Int
| [ ] => 1
| hlava :: telo => hlava * soucin telo
```

Zadání:

Naprogramujte mazání opakovaných prvků.

Řešení:

```
def vynech_opakovani {T : Type} [DecidableEq T] : List T → List T
| [ ] => [ ]
| [a] => [a]
| a :: b :: z => if a = b
                  then vynech_opakovani (a :: z)
                  else a :: vynech_opakovani (b :: z)
```

Zadání:

Naprogramujte funkci generující tzv. prefixové součty. Na pozici každého prvku bude součet čísel od začátku seznamu až po něj.

Řešení:

```
def prefixove_soucty_pom (pricist : Int) : List Int → List Int
| [ ]      => [ ]
| x :: xs => (pricist + x) :: prefixove_soucty_pom (pricist + x) xs

def prefixove_soucty : List Int → List Int :=
prefixove_soucty_pom 0
```

Zadání:

Naprogramujte funkci generující tzv. postfixové součty. Na pozici každého prvku bude součet čísel od něj až po konec seznamu.

Řešení:

Tuto úlohu lze snadno převést na předchozí:

```
def postfixove_soucty : List Int → List Int :=
obrat_rychle o prefixove_soucty o obrat_rychle
```

Jedná se o to samé jako napsat:

```
def postfixove_soucty (seznam : List Int) : List Int :=
obrat_rychle (prefixove_soucty (obrat_rychle seznam))
```

Pokud vás takové řešení neuspokojilo, můžeme také naprogramovat (poněkud komplikovanější) přímé řešení:

```
def postfixove_soucty' : List Int → List Int
| [ ]      => [ ]
| x :: xs => match postfixove_soucty' xs with
| [ ]      => [x]
| z :: zs => (x+z) :: z :: zs
```

Martin Dvořák; martin.dvorak@matfyz.cz
odevzdávejte do odevzdávátka v souboru s příponou .lean nebo .txt
(prostý text)





Axiomatizace přirozených čísel

10b

Doc.^{MM} Jana Uglickich

V tomto článku jsem se rozhodla zjistit, jak lze definovat množinu přirozených čísel a jak lze vyvodit některé její vlastnosti, hlavně tedy komutativitu a asociativitu sčítání i násobení v \mathbb{N} . Mým cílem bylo ukázat, že „nespadly z nebe“ a že, navzdory tomu, jak jsou intuitivní a často používané, nejsou zdefinované jako axiomy a naopak je možné je, právě pomocí axiomů, dokázat.

Peanovy axiomy

Peanovy axiomy definují množinu přirozených čísel a zavádí princip indukce. V současnosti se nejčastěji uvádí těchto šest:

$$1 \in \mathbb{N},^{14} \quad (1)$$

$$(a = b \wedge b \in \mathbb{N}) \implies a \in \mathbb{N}, \quad (2)$$

$$a \in \mathbb{N} \implies (a + 1 \in \mathbb{N}), \quad (3)$$

$$a, b \in \mathbb{N} \implies ((a = b) \iff (a + 1 = b + 1)), \quad (4)$$

$$\nexists a \in \mathbb{N} : a + 1 = 1, \quad (5)$$

$$(1 \in \mathbb{M} \wedge (a \in \mathbb{M} \implies a + 1 \in \mathbb{M})) \implies \mathbb{M} = \mathbb{N} \text{ (indukce)}. \quad (6)$$

Další tři axiomy, které se v dnešních materiálech už obvykle neobjevují, definují relaci „=“ na \mathbb{N} jako ekvivalenci.

Vztah $a + 1$ nazýval i Peano „následníkem a “, dnes se často zapisuje jako $S(a) = a + 1$.

Poznámka redakce: Pro pochopení celého článku je potřeba, abyste dobře rozuměli matematické indukci, která je zavedena axiomem (6). Jana indukci opakovaně využívá v důkazech, v samostatné sekci na konci článku bychom vám tedy rádi přiblížili obecný princip, podle kterého důkazy indukcí fungují, abychom vám čtení článku usnadnili.

Jana ve článku často využívá indukční předpoklad, aniž by to explicitně zmínila, většinou ve formulacích „víme, že“. To může být matoucí pro čtenáře, kteří se s indukcí ještě nesetkali, rozhodli jsme se tedy daná místa opatřit značkou ♠.

¹⁴Tento axiom může mluvit buď o 1, nebo o 0. Je možné setkat se s oběma variantami, já jsem se rozhodla zůstat u „školní“ definice přirozených čísel coby čísel vyjadřujících počet prvků neprázdné množiny.

Sčítání

Protože k přirozenému a můžeme z přirozených čísel přičíst buď 1, nebo číslo zapsatelné jako $S(b)$, definoval Peano sčítání přirozených čísel jako:

$$a + 1 = S(a), \quad (7)$$

$$a + S(b) = S(a + b). \quad (8)$$

Pomocí axiomů a této definice lze odvodit všechny vlastnosti uvedené dále: uzavřenost \mathbb{N} vůči sčítání, asociativitu a komutativitu sčítání.

Uzavřenost \mathbb{N} vůči sčítání

Chceme dokázat, že

$$\forall a, b \in \mathbb{N} : a + b \in \mathbb{N}. \quad (9)$$

Pro $b = 1$

$$a + 1 = S(a) \quad (10)$$

a z (3) víme, že $S(a) \in \mathbb{N}$. Jestliže (9) platí pro b , pak pro $S(b)$

$$a + S(b) = S(a + b).$$

Z platnosti (9) pro b plyne $a + b \in \mathbb{N}$ ♠, zatímco z (3) $S(a + b) \in \mathbb{N}$. Tím je uzavřenost \mathbb{N} vůči sčítání dokázána.

Asociativita sčítání

Chceme dokázat, že pro všechna $a, b, c \in \mathbb{N}$ platí

$$(a + b) + c = a + (b + c). \quad (11)$$

Budeme dokazovat indukcí podle c , tj. nejdřív je nutné dokázat, že tvrzení platí pro $c = 1$:

$$(a + b) + 1 = S(a + b) = a + S(b) = a + (b + 1). \quad (12)$$

Jednotlivé úpravy původního výrazu vyplynuly z (7), (8) a (7).

Tvrzení platí pro $c = 1$. Pokud platí pro c , potom pro $S(c)$:

$$(a + b) + S(c) = S((a + b) + c) = S(a + (b + c)) = a + S(b + c) = a + (b + S(c)). \quad (13)$$

Úpravy vyplynuly z (8), asociativity přičítání c ♠, (8) a (8).

Tedy pro každé c , pro které (11) platí, platí (11) i pro $S(c)$, takže platí pro všechna $c \in \mathbb{N}$. Sčítání přirozených čísel je skutečně asociativní.



Komutativita sčítání

Chceme dokázat, že pro všechna $a, b \in \mathbb{N}$:

$$a + b = b + a. \quad (14)$$

Nejprve dokážeme pro $b = 1$. Jak víme z (7), pro $a = 1$:

$$a + b = a + 1 = S(a),$$

$$b + a = b + 1 = S(b)$$

a protože $a = b$, z (4) máme $S(a) = S(b)$. Platí tedy

$$a = b = 1 \implies a + b = b + a. \quad (15)$$

Poznámka redakce: Jana má důkaz správně, ale předchozí kroky by bylo možné zjednodušit. Pokud bychom za a a b rovnou dosadili, získáme $1 + 1 = 1 + 1$, což zjevně platí, protože výraz na pravé straně je stejný jako na levé.

Když (14) platí pro $b = 1$ a a , pro $S(a)$:

$$S(a) + 1 = S(S(a)) = S(a + 1) = S(1 + a) = 1 + S(a). \quad (16)$$

Úpravy vyplynuly z (7), (7), (14) ♠ a (8).

Už jsme dokázali platnost (14) pro $b = 1$ pro všechna a . Když (14) platí pro b , pro $S(b)$:

$$a + S(b) = S(a + b) = S(b + a). \quad (17)$$

Úpravy vyplynuly z (8) a platnosti (14) ♠.

$S(b + a)$ dále upravíme:

$$S(b + a) = b + S(a) = b + (a + 1) = (b + a) + 1. \quad (18)$$

Úpravy vyplynuly z (8), (7) a (11), která byla dokázána v předchozí části.

V následujícím kroku si představíme, že „ $c = (b + a)$ “, a použijeme $c + 1 = 1 + c$, což jsme již dokázali výše v (15) a (16)::

$$(b + a) + 1 = 1 + (b + a). \quad (19)$$

Potom z (11), prozatím známých vlastností komutativity a (7):

$$1 + (b + a) = (1 + b) + a = (b + 1) + a = S(b) + a, \quad (20)$$

čímž bylo konečně dokázáno, že

$$a + S(b) = S(b) + a. \quad (21)$$

Tím je komutativita sčítání dvou přirozených čísel dokázána.

Násobení

Peano definoval násobení, z podobných důvodů jako sčítání, jako

$$1 * a = a, \quad (22)$$

$$S(b) * a = ba + a, \quad (23)$$

kde ab je jen jiný způsob zápisu $a * b$.

Uzavřenost \mathbb{N} vůči násobení

Chceme dokázat:

$$\forall a, b \in \mathbb{N} : ab \in \mathbb{N}. \quad (24)$$

Z (22) platí pro $a = 1$:

$$a * b = 1 * b = b, \quad (25)$$

a my víme, že $b \in \mathbb{N}$, takže i toto $ab \in \mathbb{N}$. Pokud (24) platí pro a , pro $S(a)$ z (23):

$$S(a)b = ab + b. \quad (26)$$

Víme, že $ab \in \mathbb{N} \spadesuit$, z (9) i $ab + b \in \mathbb{N}$. Tím je uzavřenost \mathbb{N} vůči násobení dokázána.

Distributivita násobení vůči sčítání

Chceme dokázat

$$a(b + c) = ab + ac. \quad (27)$$

Pro $a = 1$ platí z (22)

$$1 * (b + c) = b + c = 1 * b + 1 * c. \quad (28)$$

Když (27) platí pro a , pak pro $S(a)$:

$$S(a) * (b + c) = a * (b + c) + (b + c) = (ab + ac) + (b + c),$$

což platí z (23) a distributivity pro $a \spadesuit$,

$$(ab + ac) + (b + c) = ((ab + ac) + b) + c = (b + (ab + ac)) + c = ((b + ab) + ac) + c,$$

což platí z (11), (14) a (11),

$$((b + ab) + ac) + c = (b + ab) + (c + ac) = (ab + b) + (ac + c),$$

což platí z (11), (14),

$$(ab + b) + (ac + c) = S(a) * b + S(a) * c,$$

z (23) a (23).

Distributivita násobení vůči sčítání je tímto dokázána.



Komutativita násobení

Chceme dokázat, že

$$ab = ba. \quad (29)$$

Nejprve dokážeme pro $a = 1$, tedy že $1 * b = b * 1$. Z (22) víme, že vždy platí $1 * b = b$, nyní tedy musíme dokázat $b * 1 = b$.

Pro $b = 1$ platí z definice (22). Pro $S(b)$:

$$S(b) * 1 = b * 1 + 1 = 1 * b + 1 = b + 1 = S(b). \quad (30)$$

Změny vyplynuly z (23), platnosti tvrzení pro $b \spadesuit$, (22) a (7).

Teď tedy víme, že pro všechna přirozená b platí (29), ale zatím jen pro $a = 1$. Pro $S(a)$:

$$S(a) * b = ab + b = ba + b. \quad (31)$$

Úpravy vyplynuly z (23) a komutativity násobení $a \spadesuit$.

Z druhé strany:

$$b * S(a) = b(a + 1) = ba + b * 1 = ba + 1 * b = ba + b. \quad (32)$$

Úpravy vyplynuly z (7), (27), z rovnosti $b * 1 = 1 * b$ dokázané výše a (22). Teď vidíme, že

$$S(a) * b = b * S(a) = ba + b, \quad (33)$$

čímž je komutativita násobení dokázána.

Asociativita násobení

Chceme dokázat, že

$$a(bc) = (ab)c. \quad (34)$$

Pro $a = 1$:

$$1 * (bc) = bc, \quad (35)$$

což vyplývá z (22), a

$$(1 * b)c = bc, \quad (36)$$

což také vyplývá z (22).

Pro $S(a)$:

$$\begin{aligned} S(a)(bc) &= (a(bc)) + bc = ((ab)c) + bc = (c(ab)) + bc = (c(ab)) + cb = \\ &= c(ab + b) = c(S(a)b) = (S(a)b)c. \end{aligned} \quad (37)$$

Jednotlivé úpravy vyplynuly z (23), platnosti tvrzení pro $a \spadesuit$, (29), (29), (27), (23) a (29).

Zdroje

- [1] Peano, Giuseppe: Arithmetices principia, nova methodo exposita, <https://archive.org/details/arithmeticespri00peangoog>
- [2] Nahas, Michael: Peano_book, https://github.com/mdnahas/Peano_Book
- [3] Wallach, Nolan: Peano Axioms, <https://mathweb.ucsd.edu/~nwallach/peano.pdf>

Poznámka redakce: Matematická indukce

V první části článku je zavedena matematická indukce axiomem

$$(1 \in \mathbb{M} \wedge (a \in \mathbb{M} \implies a + 1 \in \mathbb{M})) \implies \mathbb{M} = \mathbb{N}.$$

Pojďme se na ni nyní podívat blíže. Pro intuitivní představu si ji ilustrujeme příkladem padajícího domina.

Představme si, že máme kostky domina postavené v řadě za sebou a očíslované přirozenými čísly. Víme, že pokud spadne kostka domina a , určitě spadne i kostka $a + 1$. Nyní necht' první kostka domina spadne. Víme, že určitě spadla i druhá, protože stála hned za ní. Když spadla druhá, tak určitě spadla i třetí. Podobný argument můžeme používat stále dokola, až projdeme všechny kostky a o všech víme, že spadly. To je hlavní princip matematické indukce. Pokud víme, že pro nějaké přirozené číslo tvrzení platí, a že pokud platí pro jedno, pak určitě musí platit i pro to následující, vyplývá z toho, že musí platit pro všechna. Jak si to ale představit v kontextu axiomu výše?

Řekněme, že \mathbb{M} je množina spadlých kostek.

- víme, že první kostka spadla, máme tedy *základ indukce*:

$$1 \in \mathbb{M}$$

- pokud spadla kostka a (tedy platí *indukční předpoklad* $a \in \mathbb{M}$), víme, že spadla i kostka $a + 1$, což nám dává *indukční krok*:

$$a \in \mathbb{M} \implies a + 1 \in \mathbb{M}$$

- z toho podle axiomu vyplývá, že spadnou všechny kostky, tedy

$$\mathbb{M} = \mathbb{N}$$

Není složité ukázat, že pokud spadne první kostka domina v řadě, spadnou i všechny ostatní. Pojďme si nyní ukázat obecný princip důkazu indukcí na jednom z tvrzení, které uvádí Jana ve svém článku, konkrétně na uzavřenosti \mathbb{N} vůči sčítání.



Chceme dokázat, že $\forall a, b \in \mathbb{N} : a + b \in \mathbb{N}$, tedy pro libovolná a, b přirozená je i jejich součet přirozené číslo.

Budeme dokazovat indukcí podle b . To znamená, že nejprve ukážeme, že pro $b = 1$ tvrzení platí (*základ indukce*), a potom, že pokud platí pro b (*indukční předpoklad*), tak z toho plyne, že platí i pro $b + 1$ (*indukční krok*). V našem případě důkaz provedeme nezávisle na tom, jaké máme a , tvrzení tedy pak platí pro libovolné a , jinak řečeno pro všechna a .

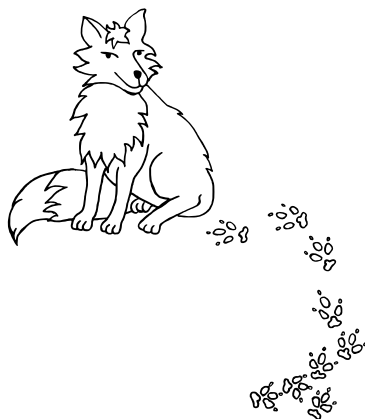
Nejprve vždy potřebujeme *základ indukce*. V tomto případě musíme tedy ukázat, že pro $b = 1$ platí $a + b \in \mathbb{N}$. Pro $b = 1$ máme $a + b = a + 1$, $a + b$ je tedy vlastně následník a , kterého můžeme jinak označit jako $S(a)$. Z axiomu (3) víme, že následník přirozeného čísla je přirozené číslo, tedy $S(a) \in \mathbb{N}$. *Základ indukce* tedy máme, ukázali jsme, že pro $b = 1$ tvrzení platí.

Nyní přejdeme k *indukčnímu kroku*. Ten obvykle provádíme tak, že předpokládáme, že tvrzení platí pro b , v tomto případě tedy $a + b \in \mathbb{N}$. To je náš *indukční předpoklad*. Následně chceme dokázat, že v takovém případě platí i pro $b + 1$, tedy následníka b označeného $S(b)$. Formálně $a + S(b) \in \mathbb{N}$.

Z definice sčítání víme, že $a + S(b) = S(a + b)$. Z indukčního předpokladu plyne $a + b \in \mathbb{N}$ – předpokládáme prostě, že pro b tvrzení platí. Když si $a + b$ označíme jako c , tak z (3) hned vidíme, že $S(c) \in \mathbb{N}$, tedy $S(a + b) \in \mathbb{N}$. Tím je uzavřenost \mathbb{N} vůči sčítání dokázána, součet libovolných dvou přirozených čísel je přirozené číslo. \square

Pro připomenutí to ještě jednou zopakujeme: každý důkaz indukcí musí obsahovat dvě části, základ indukce a indukční krok. I když jsou třeba jednoduché, musí tam být oba, jinak důkaz nefunguje! Častou chybou je zapomenout na základ indukce, ale to je jako byste chtěli ukázat, že všechny kostky domina spadly, ale nestrčili jste do té první.

Káťa Č.; katerrina98@gmail.com



Téma 3 – Hex

K prvnímu deadline druhého čísla nám mnoho řešení nepřišlo, a proto naše hlavní problémy zůstávají stejné jako v předchozím díle. Uzavřeme však Úlohu 2.1 a Problém 2.5 jejich řešeními a ukážeme si (úplně na závěr tématka) vítěznou strategii Dr.^{MM} Martina Těšitele pro plochu 7×7 , která je moc hezky popsána (kdybyste v ní však našli nějakou díru/chybu, neváhejte o ní napsat nejlépe rovnou i s opravou této chyby).

Řešení druhého (a trochu i prvního) dílu

Úloha 2.1

Zadání:

Ukažte, že v základní hře (11×11) má vyhrávající strategii červený hráč (začínající), neboli že když budou hrát hráči perfektně, vyhraje červený. Bonusové body dostanete, pokud o svém správném důkazu řeknete co nejobecněji, pro jaké hrací plochy platí.

Připomínáme, že vyhrávající strategie pro 11×11 není známa a hra byla oblíbená mezi mnoha matematiky, takže řešení popisující konkrétní strategii bude téměř jistě špatně.

Řešení:

Důkaz provedeme sporem: představme si, že vyhrávající strategii má nezačínající (modrý) hráč (remízou základní hra skončit nemůže, jak už jsme ukázali v minulém díle). Červený pak ale může v prvním tahu zahrát cokoliv, tento první svůj kámen ignorovat, a převzít tak roli nezačínajícího hráče. Takže červený vyhraje, ať hraje modrý cokoliv včetně své vyhrávající strategie, což je spor (oba nemůžou mít vyhrávající strategii).

Proč to ale stačí? (Proč třeba nemůže nastat nevýhoda tahu, jako v šachách nebo dámě, tedy situace, kdy se nezačínající hráč může přizpůsobit prvnímu tahu začínajícího hráče a vyhrát?) Červený tedy v prvním tahu někam položí kámen (dejme si na něj značku X) a dále hraje přesně strategii nezačínajícího hráče jako by tam kámen s X nebyl. Vždy, když mu strategie nezačínajícího hráče příkazuje hrát na políčko s X , zahraje na libovolné volné políčko, a přesune značku X tam. Navíc vzhledem k pravidlům kámen s X nedává žádnou novou možnost výhry modrému a naopak červenému žádnou neubírá. Tedy červený opravdu může „hrát strategii nezačínajícího hráče“ a vyhraje nejpozději tehdy, kdy by měl vyhrát podle této strategie, tedy „dřív než modrý“.

Nezapomínejte, že i druhá část je součástí důkazu a bez ní je první část v podstatě k ničemu.

Řešení bonusové části:

Tento důkaz platí pro všechny symetrické plochy. Symetrické ve smyslu toho, že když na hrací ploše prohodíme červenou a modrou, dostaneme tu samou plochu, jen pootočenou / osově zrcadlenou.



Problém 2.5

Zadání:

3 z modifikací (počítáno jedna hranatá závorka = jedna modifikace) samy o sobě nedělají Hex o moc zajímavější, s některými dokonce hra pozbude smyslu. Víte, které máme na mysli? A proč?

- Místo tahu by hráč mohl odebrat soupeřův kámen.
Varianta: jen odebere a hraje soupeř.
- Pokud hráč obkrouží kameny soupeře, stávají se tyto kameny jeho.
- Hráči mohou obarvovat pouze pole, se kterými sousedí jejich barva.

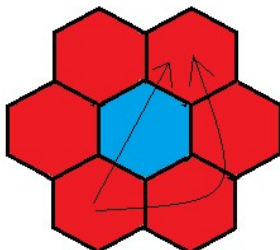
Řešení od Dr.^{MM} Martina Těšitele

Obkroužení kamenů: Jestliže jsem plně obkroužil soupeřovy kameny, nejsou mu už k ničemu, protože nespojují žádné z jeho dalších kamenů a jejich přebarvení mu tak nijak neškodí.

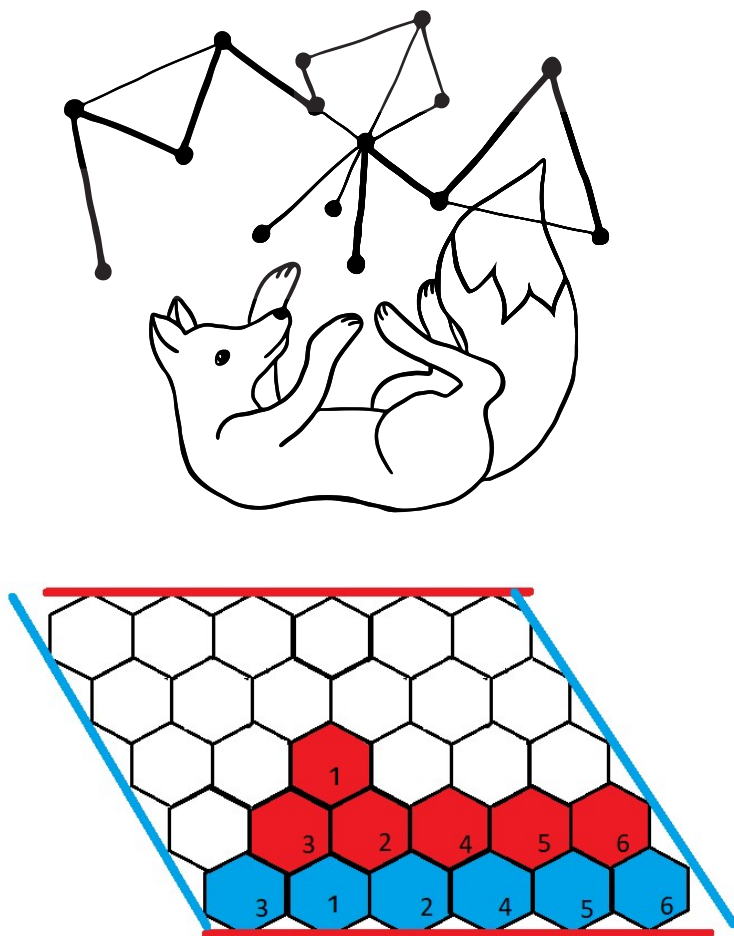
Mně ovšem také nepomůže, protože tyto nové kameny nespojují žádné z mých již nespojených kamenů (viz Obrázek 2).

Pokládání kamenů jen na pole sousedící s jinými hráčovými kameny: Jestli jsem tuto modifikaci pochopil správně, tak podle ní hráč musí pokládat své kameny jen na políčka, která sousedí s jinými jeho kameny.

Předpokládám, že první tah může udělat každý z hráčů na libovolné pole. Dejme tomu, že červený chce propojit horní a dolní a modrý levou a pravou stranu. V tom případě zahraje červený na libovolné pole (např. do středu hrací plochy) a modrý půjde pod/nad něj do vzdálenosti min. 2 pole. Dochází zde k opozici a modrý podobně jako v šachách nepustí červeného ve svislém směru, přičemž se bude sám rozrůstat do vodorovného směru a vyhraje (viz Obrázek 3)



Obrázek 2: Obkroužení polí



Obrázek 3: Pokládání kamenů jen na pole sousedící s jinými hráčovými kameny

Řešení:

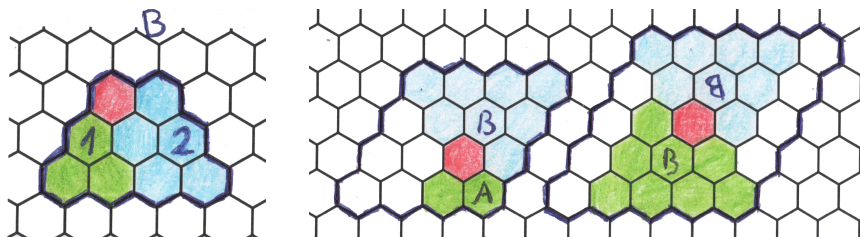
Je jedno, jak jste úpravu se sousedícími kameny pochopili, vždy úprava moc omezuje možnosti hráčů a tak má jeden z nich přímočarou vyhrávající strategii.

Při SWAPu sice také víme, kdo má vyhrávající strategii, ale pořád je dostatečně komplexní (asi i více než u původní hry), tj. zajímavá na hru. Ten jsme nemysleli jako modifikaci nepřidávající nic navíc.

Mysleli jsme modifikaci, že hráč může odebrat soupeřův kámen. V takovém případě se totiž hra zacyklí, neboť protivrác ho tam může hned vrátit. A jestliže žádný z hráčů nechce prohrát, tak vždy odebere kámen tak, aby ten druhý nevyhrál. Takže taková hra není moc hratelná.

Mohlo by se zdát, že podobné modifikace (např. místo odebrání změním soupeřův kámen na náš) také nedávají smysl,¹⁵ ale můžeme si všimnout, že tímto způsobem může hráč vyhrát (změnou se vytvoří vítězné spojení stran), takže to druhý hráč už nemůže vrátit zpět.

Další řešení



Obrázek 4: Obrázek od Dr.^{MM} Veroniky Menšíkové s řešením k části problému 2.2: Jednoduchý 1 bod získáte, když naleznete (a dokážete) větší pattern, kterým pak jednoduše popíšete vítěznou strategii na 5×5 a spolu s dvojhopem i 4×4 . Na levém obrázku je důkaz, že je pattern vyhraný za červeného (má-li již obsazené levé horní políčko a dolní okraj), jelikož modrý může zabránit výhře pouze v jedné části (postavením na políčko 1 vytvoří červený dvojhop v levé části, postavením na políčko 2 vytvoří dva v pravé). Pravý obrázek pak ukazuje, jak pattern využít k důkazu výhry na 4×4 a 5×5 .

Z druhého dílu nám Dr.^{MM} Martin Těšitel vyřešil, za koho je SWAP vyhraný. Samozřejmě je to za vybírajícího hráče (a tedy SWAP 2 se v ideálním případě nebude hrát), protože každé rozestavení SWAPu musí být za jednoho hráče vyhrané a toho si vybírající hráč vybere.

Ještě nám k prvnímu dílu přišlo pár zajímavých modifikací hry (modifikace z minulého dílu tu znovu neuvádíme):


- Hráči hrají současně. Oba naráz zvolí políčko, kam chtějí položit svůj hrací kámen. Pokud zvolí stejné políčko, zahrájí si o něj kámen nůžky papír, nebo si hodí mincí, ... [Mgr.^{MM} Barbora Vosáhllová]
- Hrací plocha má tvar rovnostranného trojúhelníku, kdy cílem obou hráčů je propojit všechny tři strany. [Mgr.^{MM} Barbora Vosáhllová] (Tato modifikace je hra se jménem Y a můžete ji znát z AZ-kvízu.)
- Další možnou úpravou pravidel je, kdy hráči navzájem neví, co zahrál protihráč. Zjistí to jen v momentě, kdy se pokusí zahrát pole, které už má zabrané protihráč. [Mgr.^{MM} Barbora Vosáhllová] (Potřebuje PC, či jiný způsob externí kontroly.)


¹⁵A nekonečné prohazování vlastně také může nastat, ale to se dá odstranit např. omezením počtu výměn nebo tím, že každé pole jde změnit jen dvakrát.

- Některá políčka označit. Pro získání těchto bude muset hráč zabojovat (hodem kostkou, v Piškvorkách s protihráčem, atd.) jinak ho získá soupeř. [Bc.^{MM} Matouš Rybecký]
- Změna hry na závody: na předem označená políčka mohou vstoupit všichni a vyhrává ten, kdo první vytvoří „cestu“ přes všechna tato pole. [Bc.^{MM} Matouš Rybecký]
- Kombinace s Piškvorkami: Jakmile někdo zahraje 3 (nebo 4 nebo 5) v řadě, hraje ještě jednou (řady mezi sebou sdílí nejvýše jedno pole). [Dr.^{MM} Veronika Menšíková]


Díl 3: Pokračování zkoumání

V tomto díle tedy nebude nic nového, jen zopakovaná zadání:

Problém 2.2: *Nalezněte herní plochy, na kterých umíte ukázat konkrétní vítěznou strategii za jednoho z hráčů. (Samozřejmě mimo těch, které byly již ukázány.) Inspirovat se můžete již ukázanými patterny.* 

Problém 2.3: *Nalezněte herní plochy, kde má modrý hráč tak velkou výhodu, že vyhraje, ať bude červený hrát jakkoliv, přestože nezačíná. (Takové, které ještě nebyly ukázány. Tj. u kosodélníků hledáme něco mezi „dvojnásobkem“ a „jednásobkem“.)* 

Můžete například zkusit zjišťovat, pro které rozměry kosodélníků vyhrává červený a pro které vyhrává modrý. (Není nutné toto rozhodnout hned pro všechny, ale např. „Když to bude mít modrý 42krát blíže, tak už vyhraje, a když to bude mít o jedna blíže, tak vyhraje červený.“¹⁶ Nic mezi nevím.“ Ale samozřejmě se boduje, které všechny kosodélníky umíte rozhodnout.)

Problém 2.4: *Samozřejmě stále můžete posílat zajímavé nápady na modifikace (které zde ještě nebyly uvedeny). Ale budeme radši, když si nějakou modifikaci vezmete a zanalyzujete (například jestli stále vyhraje červený hráč, pokud budou oba hrát perfektně, k čemuž vám může dopomoci bonus v první úloze, nebo jestli pořád musí některý z hráčů vyhrát, ať hrají hráči jakkoliv).* 

Těšíme se na vaše řešení. Doporúčujeme se zaměřit hlavně na patterny.

Bětka; betka.n@centrum.cz
Jidáš; jonas.havelka@volny.cz
odevzdávejte do odevzdávátka

¹⁶Toto tvrzení nemusí být pravda, nevím, kdo vyhraje, když to bude o jedna.

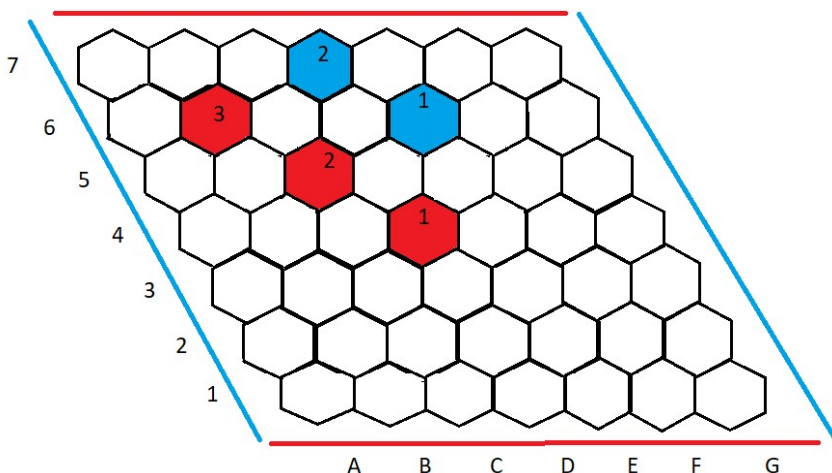
Vyhrávající strategie pro hex 7×7

4b

Dr.^{MM} Martin Těšitel

Červený začne přesně uprostřed hracího pole (D4). Modrý má nyní několik možností:

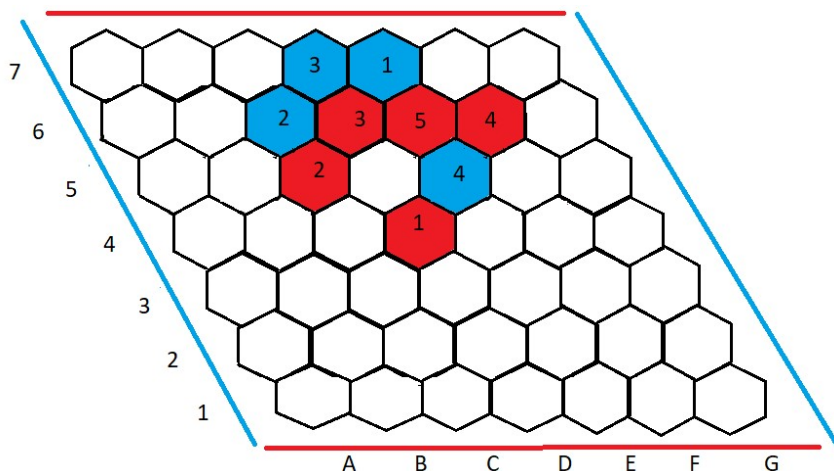
A) Jít přímo do opozice. Pokud půjde modrý přímo do opozice, (tj. na pole E6) můžeme si všimnout, že ponechává červenému více prostoru na levé části hracího pole. Červený proto zahraje na B5 a vytvoří hrozbu D7. Po tomto tahu by červený zaručeně spojil střed s horní stranou a stejný postup by mohl opakovat i pro spodní polovinu hracího pole, takže modrému nezbývá nic jiného, než aby na D7 zahrál sám. Poté však červený může zahrát na B6 a spojení středu s horní stranou bude nevyhnutelné. To samé nastane i ve spodní půlce pole.



Obrázek 5: Možnost A

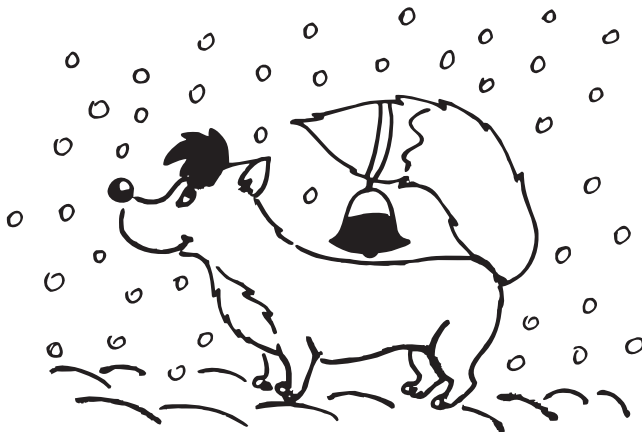
B) Vybudovat obranu na 7. řadě. Modrý nemusí červenému bránit ihned. Je možné vybudovat pevnou obranu na poslední řadě, kterou nebude moci červený obejít tak lehce jako v případě A.

Na obrázku 2.2 zahrál modrý na pole E7, ve snaze být co nejmí ve středu řady. Nyní hraje červený, a to na pole C5, aby na modrém vynutil C6 a mohl pokračovat na D6 opět s vynuceným tahem, tentokrát na D7. Těmito zdánlivě nesmyslnými tahy vytvořil červený silný střed, který nebude problém spojit s horní stranou po tahu F6, kdy modrý nemůže tomuto spojení zabránit ani tahem E5 pro následné E6. Stejný scénář se bude opakovat i ve spodní polovině hracího pole.



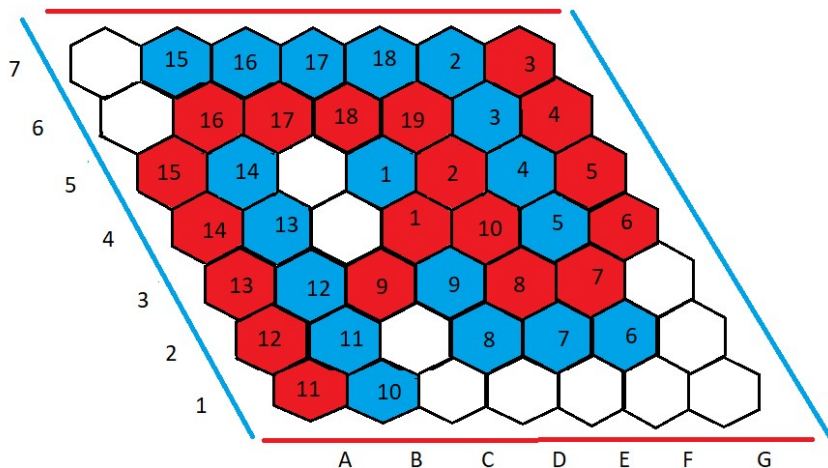
Obrázek 6: Možnost B

C) Nalepení se na červeného, aneb odříznutí od polí D5 – A7. Tato obrana mi přišla jako nejúčinnější a měl jsem poměrně problém ji prorazit. Po modrého tahu D5 se totiž červený nedostane na pole D5 – A7, přičemž jediné tahy, které ho přibližují k poslední řadě jsou E5, nebo F5. Po F5 ale přichází F6 a cesta k poslední řadě je navždy uzavřena. Zbývá tedy pouze E5, po kterém přijde F7 a po červeného F6 se modrý hrozí dostat na G7 a opět červeného od poslední řady odříznout. E6 nepomáhá pro E7, a tak je posledním možným tahem právě G7, které jako jediné drží kontakt s poslední řadou, ale dostává červeného do rohu. Modrý pochopitelně červeného odřízne tahem F6 a jediná možnost červeného, po které nebude od poslední řady odříznut tahem G6, je sám G6 zahrát, načež





ho modrý nebude chtít pustit dál a bude pokračovat tahem F5. Celý cyklus se opakuje do okamžiku, kdy se červený dostane na pole G3 (pokud modrý červeného pustí dříve, než se dostane ke G3, červený využije svého středu k předběhnutí modrého blokády. Viz Obrázek 7). To už musí modrý zahrát na F1, protože po



Obrázek 7: Uvolnění cesty červenému před bodem G3 (možnost C)

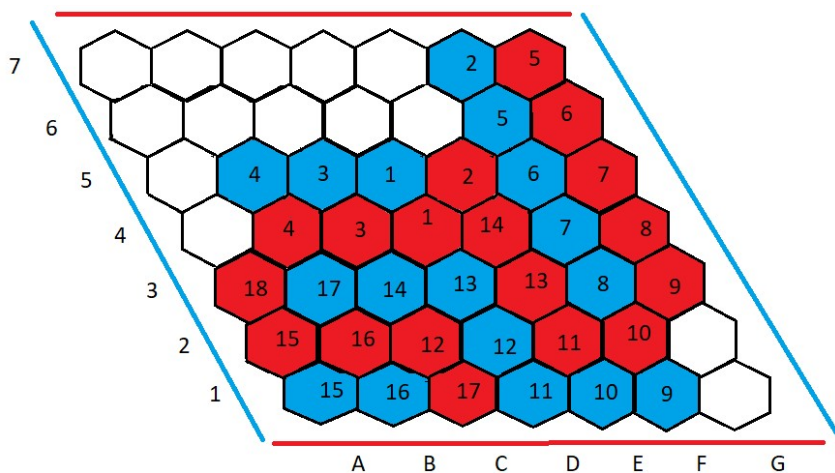
F2, G2 (případně G2, F2) musí červený spojit své strany. Červený musí zahrát F2, protože jinak by byl opět odříznut od zbytku hrací plochy, načež musí modrý jít na E1 a zabránit tak červenému ve výhře. Po dalším v podstatě vynuceném E2, D1 musí červený zahrát C2, protože jinak nemá šanci dostat se přes bariéru modrého. Modrý musí krýt hrozbu D2, tím, že D2 zahraje sám, tím je vynuceno červeného E3 (brání odříznutí od poslední řady) a modrého D3 (opět brání spojení stran). Nyní musí červený zahrát na E4 a dovolit modrému ho úplně odříznout tahem C3.

Tato pozice je prohraná, ale nebyla by, kdyby měl červený obsazené body C4 a B4, na což už nezbyvá čas. Nebo snad ano?

Červený totiž jako svůj třetí tah vůbec nemusí hrát G7, ale může obsadit C4 s hrozbou C5, kterou musí modrý bránit! Nejlepší možná obrana je, když C5 obsadí sám, načež červený obsadí pole B4 a celý proces se zopakuje.

Nyní má červený svá pole C4 a B4, a tak uskuteční úplně stejné tahy jako minule. Poté už stačí po modrého C3 zahrát na A2 a výhra je na první pohled patrná. Modrému už nepomůže ani A1, pro B2, které zaručuje kontakt s první řadou. Nejlepší strategie je tedy: D4–D5, E5–F7, C4–C5, B4–B5, G7–F6, G6–F5, G5–F4, G4–F3, G3–F1, F2–E1, E2–D1, C2–D2, E3–D3, E4–C3, A2–A1, B2–B1, C1–B3, A3 1–0

Pozn. Jsou zde i krátké, nebo velmi jednoduché odbočky, které jsem nerozepsal a je samozřejmě možné, že jsem nějakou obranu modrého přehlédl, ale toto



Obrázek 8: Nejlepší strategie červeného po dokonalé obraně modrého (možnost C s obsazením polí B4 a C4)

je pravděpodobně nejlepší možná strategie pro oba hráče. Pokud někomu tento text přijde příliš zdlouhavý, nebo umělecký tak se omlouvám, ale vycházím ze šachové literatury, kde jsou takové texty naprosto běžné.





Téma 4 – Genetika

Vítáme vás u dalšího čísla témátka genetiky. Tentokrát se nepustíme do žádného nového tématu, naopak bychom vám rádi připomněli otevřené problémy, které jsme zadali v druhém čísle. Budeme moc rádi, pokud se nad danou problematikou zamyslíte a napíšete nám o ní článek. O tom, jak napsat článek, se můžete dočíst na našich stránkách.



Problém 2.5: *Tento díl alespoň stručně shrnoval historii genetiky. Jaké další chvíle byly významné a jaký má tento obor vůbec význam pro společnost?*

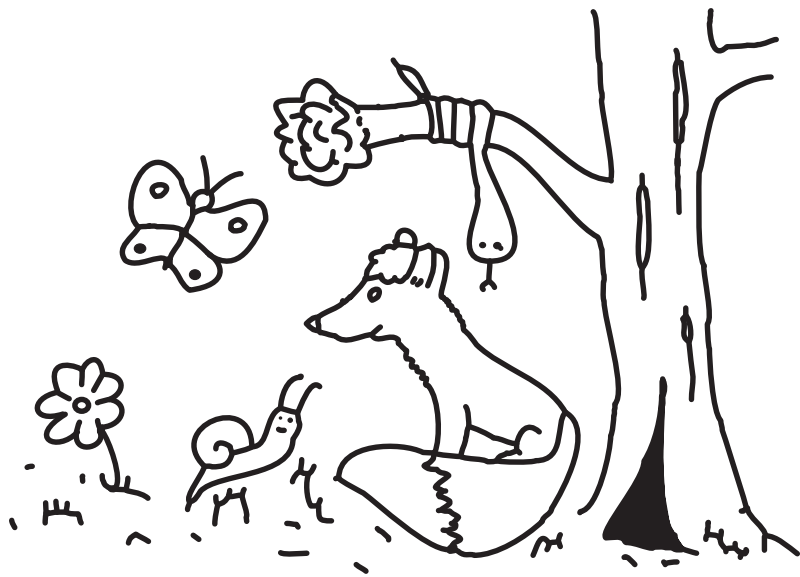


Problém 2.6: *Představte si, že jste v kůži Mendela a chcete znovuobjevit tři pravidla. Jak byste postupovali? S jakým živočichem, rostlinou nebo na jakém neživém předmětu byste to ukázali? (Představivosti se meze nekladou, tak se vyhněte hrachu.)*

Juli; julie.krimska@mensa.cz

Fofík; martin.fof1@seznam.cz

odevzdávejte do odevzdávátka



Problém klikujícího hráče výzvy

7b

Mgr.^{MM} Jáchym Löwenhöffner

Představme si následující problém. Užíváte si sous, když v tom přijde váš kamarád, kterého jste alespoň hodinu neviděli, a prohlásí „Dej si deset“. ¹⁷ Tedy pokleknete k zemi a začnete klikovat. Abyste náhodou neudělali více než deset kliků, tak si je nahlas počítáte.

1...2...3...výzva! Váš kamarád stojí nad vámi a je připraven při každé příležitosti říct výzvu. ¹⁸ Přijmete svůj osud a když už jste v tom klikování, tak si za každou výzvu dáte prostě jeden navíc.

Kolik kliků budete dělat?

Pro případ *dej si deset* je to víceméně triviální. Výsledně uděláte 14 kliků. Protože za prvních deset dostanete tři výzvy a poté až za dvanáctý klik. Jednoduché intuitivní řešení je prostě počet kliků vydělit třemi a výsledek přičíst. Tím ovšem zapomínáme na výzvy, které dostaneme při dělení kliků z první vlny výzvy.

Když n_0 je počet kliků, které jste dostali, tak předchozí řešení bude fungovat jen pro $\lfloor n_0/3 \rfloor + (n_0 \bmod 3) < 3$. My ovšem chceme obecné řešení. Můžeme si říct, že v každém kroku přidáme rozdíl předchozích dvou vrstev děleno třemi. Jako vrstvu vnímáme každou chvíli, kdy bychom bez dalších výzev skončili. Tedy když $n_0 = 10$, tak první vrstva končí v kliku 10, druhá v kliku 13 (dále psáno jako $n_1 = 13$) a poslední vrstva končí v kliku 14, tedy $\Delta n_2 = 1$. Předchozí výraz můžeme číst také jako: mezi první a druhou vrstvou musíme udělat jen jeden klik navíc.

I když význam Δn_i je rozdíl mezi dvěma vrstvami kliků, zapisujeme ji možná trošku nelogicky, a to:

$$\Delta n_i := \left\lfloor \frac{n_{i-1}}{3} \right\rfloor - \left\lfloor \frac{n_{i-2}}{3} \right\rfloor$$

To nám otevírá další možnost jak nad Δn_i přemýšlet a to jako o rozdílu kliků, které bych měl přidat za celou dobu, a kliků, které jsem již přidal a udělal. Od začátku bych měl přidat $\lfloor n_{i-1}/3 \rfloor$, ale z nich jsem již udělal $\lfloor n_{i-2}/3 \rfloor$.

V i -té vrstvě tedy budeme mít za sebou n_i kliků.

$$n_i = \Delta n_i + n_{i-1}$$

Když si n_{-1} definujeme jako 0, tak ve výsledku musíme udělat $\mathcal{K}(n_0)$ kliků:

$$\mathcal{K}(n_0) := n_0 + \sum_{i=1} \Delta n_i$$

Sumu sčítáme až do $\Delta n_i < 1$.

¹⁷Tato hláška je součástí hry, kde každý zapojený může dát komukoliv jinému zapojenému hráči deset kliků. Jednotlivé hlášky „dej si deset“ ale musí být alespoň hodinu od sebe.

¹⁸Další podobná hra, kde jde ovšem o to, že pokaždé, kdy řeknete číslo dělitelné třemi a někdo další vám řekne výzvu, musíte se dotknout nosem země.



Zajímavé pozorování můžeme udělat, když se podíváme, jakých hodnot může \mathcal{K} nabýt. Jsou to všechna čísla, která nejsou dělitelná třemi. Protože kdybychom měli skončit na kliku dělitelném třemi tak si uděláme ještě jeden navíc. Dobré je si taky rozmyslet, že \mathcal{K} je funkce prostá. Tedy pro každé n_0 uděláme ve výsledku jiný počet kliků. Toto pozorování můžeme zapsat následovně¹⁹:

$$\mathcal{K} : \mathbb{N} \mapsto \{x \mid x \bmod 3 \neq 0, x \in \mathbb{N}\}$$

Stihnete to do hodiny?

Dalším dobrým dotazem je, jestli stihnete doklikovat do jedné hodiny. Jestli ne, kamarád by vám mohl dát dalších deset a už byste v klikování pokračovali na vždy. Jako v předchozí části, varianta pro deset kliků je triviální. Každý asi zvládne udělat deset kliků do jedné hodiny. Co kdybyste ale dostali třeba 100 kliků?

Důležitým předpokladem v této úvaze je, že každý další klik je exponenciálně těžší než ten předchozí. Tento předpoklad dává smysl, protože s každým klikem jsme více a více unaveni.²⁰

Protože každý člověk má jinou fyzičku a jinou výdrž při děláni kliků, nedá se obecně určit vztah počtu už udělaných kliků k času dalšího kliku. Konstanta c vyjadřuje přesně toto. Celá funkce $\mathcal{T}(x)$ vrací počet sekund, kolik nám zabere x -tý klik (znamenáje, že jsme jich už udělali $x - 1$).

$$\mathcal{T}(x) := e^{cx}$$

Abychom zjistili, kolik času nám tedy dohromady zabere $\mathcal{K}(n_0)$ kliků, musíme spočítat sumu funkce \mathcal{T} od nuly až do $\mathcal{K}(n_0)$.

$$\sum_{x=0}^{\mathcal{K}(n_0)} \mathcal{T}(x)$$

Když je tato suma větší než 3 600 (počet sekund v hodině), tak víme, že tolik kliků rozhodně do hodiny nestihnete. Jestli vás zajímá pro jaké n_0 už výsledný počet kliků nestihnete pod hodinu, pokuste se experimentálně změřit konstantu c a poté dopočítat maximální hodnotu n_0 . Na spočítání $\mathcal{K}(n_0)$ doporučuji použít program.

¹⁹Množina napravo jsou všechna přirozená čísla nedělitelná třemi. Šipka mezi množinami se používá na znázornění prostého zobrazení.

²⁰Zajímavé je, že část pochodů v lidském těle se chová podle nějaké exponenciály. Například vnímání intenzity světla, hladiny zvuku atd.

Poř.	Jméno	R.	\sum_{-1}	Témata				Ostatní	\sum_0	\sum_1
				1	2	3	4			
37.	A. Weberová	4	11,6	1,0	6,6	4,0		11,6	11,6	
38.	Mgr. ^{MM} O. Popovský	4	99,2			0,3		0,3	11,5	
39.	Dr. ^{MM} V. Menšíková	2	140,1			11,3		11,3	11,3	
40.	P. Barták	Z9	9,9						9,9	
41.	Dr. ^{MM} J. Tregler	4	116,3		9,4			9,4	9,4	
42.	Mgr. ^{MM} M. Ulumbekov	2	70,4						8,7	
43.	A. Stýskala	4	13,9						8,4	
44.	Bc. ^{MM} V. Kučera	2	41,4		2,3		1,6	3,9	7,8	
45.	A. Migel	1	7,3		7,3			7,3	7,3	
46.	Bc. ^{MM} V. Vybíral	2	48,7	2,0		4,3		6,3	6,3	
47.	Dr. ^{MM} D. Kaňka	2	122,6	4,0	0,5	1,0		5,5	5,5	
48.	Bc. ^{MM} V. Verner	3	48,9	4,0				4,0	4,0	
49.	R. Zelený	3	3,8						3,8	
50.–51.	L. Trochová	1	3,0		0,5	2,5		3,0	3,0	
	Bc. ^{MM} R. Petit	3	37,0						3,0	
52.	Bc. ^{MM} K. Menšíková	1	24,9	0,5		1,4		1,9	1,9	
53.	Bc. ^{MM} M. Rybecký	2	42,9			1,7		1,7	1,7	
54.	S. Teodorovičová	3	8,1						1,0	

Sloupec \sum_{-1} je součet všech bodů získaných v našem semináři, \sum_0 je součet bodů v těchto deadlinech a \sum_1 součet všech bodů v tomto ročníku. Tituly uvedené v předchozím textu slouží pouze pro účely M&M.



Časopis M&M je zastřešen Matematicko-fyzikální fakultou Univerzity Karlovy. S obsahem časopisu je možné nakládat dle licence CC BY 4.0. Autory textů jsou, není-li uvedeno jinak, organizátoři M&M. Realizace projektu byla podpořena Ministerstvem školství, mládeže a tělovýchovy. Pokud si časopis nepřejete dále dostávat v tištěné podobě, zrušte si prosím jeho odběr v nastavení svého účtu na webu.

Kontakty:

M&M, OPMK, MFF UK E-mail: mam@matfyz.cz
 Ke Karlovu 3 Web: mam.matfyz.cz
 121 16 Praha 2 FB: [casopis.MaM](https://www.facebook.com/casopis.MaM)

