

JARNÍ SOUSTŘEDĚNÍ KSP 2012 – Seznam Přednášek

Tento spisek jest nabídkou přednášek, které byste na soustředění mohli slyšet, čili jakási obdoba matfyzácké Karolínky (ta je ale, pravda, ještě stále o něco tlustší). Přednášek je daleko víc, než kolik se dá za pár dní stihnout, a tak je na vás, abyste si vybrali, o které máte opravdu zájem; pokud byste rádi slyšeli ještě o něčem dalším, klidně to k přednáškám připište, třeba se najde někdo, kdo by vám o tom rád pověděl. Berte a vychutnávejte!

Údaje o jedné přednášce vypadají asi takto:

Stručný úvod do základů teorie vlkodlaků (“*Za dne ukryt v hloubi lesa, děs temný zvečera se plazí...*”) **LYK**

RNDr. Á. Cula

Úvod do moderní teorie vlkodlaků, čili též praktická daemologie a naiadologie.

Předpoklady: Měsíc v úplňku.

Dozvíte se (čteno v obvyklém pořadí): jméno přednášky, v uvozovkách motto přednášky, kód (pro snadnější odkazování na konkrétní předměty), jméno přednášejícího a nakonec stručný obsah přednášky.

Povinně dobrovolné přednášky

Základy programování (“*Má $x = x + 1$ řešení?*”) **ZAKL**

Radim „Rumcajz“ Cajzl, Jirka Setnička

Úvodní několikadilná přednáška pro ty, kteří mají s programováním malé nebo dokonce žádné zkušenosti. Od základů si vysvětlíme problematiku programování a představíme programovací jazyk Python, který je jednoduchý na naučení, ale přitom není jen dětskou hračkou a je používán ve spoustě důležitých projektů. Ukážeme si základní datové typy (n-tice, seznamy, slovníky) a datové struktury (fronta, zásobník) a základní algoritmy a programátorské techniky, bez kterých se žádný programátor neobejde.

Základní algoritmy **ZALG**

Základní výbavou každého informatika jsou různé standardní algoritmy, zde si ukážeme ty nejdůležitější z nich: Třídící algoritmy včetně vnějšího třídění. Vyhledávání v polích, hledání mediánu, resp. k -tého největšího prvku v lineárním čase. Vyhodnocování výrazů, předzpracování vstupních dat.

Grafy & algoritmy I (“*Pojďme si hrát s obrázky*”) **GA1**

Co to jsou grafy, jak je v programech reprezentovat a hlavně k čemu se dají použít. Prohledávání grafu do šířky i do hloubky. Hledání nejkratších cest: Dijkstrův a Floydův algoritmus. Topologické třídění grafů a kreslení grafů jedním tahem.

Datové struktury pro začátečníky **DS1**

Jak si ukládat data natolik šikovně, abychom je nejen neztratili, ale také našli dříve, než si pro nás přijde Smrt. Klasické struktury jako pole, seznamy, vyhledávací stromy (vyvážené, AVL, a - b , splay), haldy (binární a obecně regulární) a v neposlední řadě hashování.

Algoritmy a jejich složitost (“*Čím menší je časová složitost algoritmu, tím větší je složitost kódu.*”) **SLOZ**

Problém, algoritmus a program. Časová a paměťová složitost problémů i algoritmů. Složitost rekurzivních algoritmů, složitost v průměrném případě.

Dokazování **DOKAZ**

Radim „Rumcajz“ Cajzl

Ověřit správnost programu můžeme buď spuštěním algoritmu na opravdu velkém množství různorodých vstupů, nebo rozumovou úvahou, která mimo vši pochybnost existenci vstupu, na který by program odpověděl špatně, vyloučí.

Dynamické programování (“*Kampak jsem si to jenom schoval?*”) **DYNP**

Karel Tesař, Jirka Setnička

Dynamické programování je programátorská technika využívající velice prostinkého nápadu: Proč něco počítat několikrát, když to mohu spočítat jednou a výsledek si uložit? Na této přednášce si ukážeme, že tento jednoduchý nápad může pomoci efektivně vyřešit i poměrně obtížné úlohy.

Jak vypadá řešení (“*Jen jeden bod, když jsem napsal 18 stránek?*”) **SOL**

Jak má správně vypadat řešení KSP? Na co si dát pozor, co je úplně špatně a za co organizátoři strhávají body a sobě vlasy. Přednáška, která by mohla pomoci i mnohým déle aktivním řešitelům.

Algoritmizace

Složitější složitost * **SLOZ2**

Martin Böhm, Martin Mareš

Trochu hlouběji o složitosti: amortizovaná časová složitost, dolní odhady, nedeterministické výpočty a třída NP, NP-úplné problémy a příklady redukci.

Předpoklady: SLOZ

Datové struktury pro pokročilé *

DS2

Martin Böhm, Karel Tesař, Martin Mareš

Důmyslnější datové struktury: trie, splay stromy, BB- α stromy; geometrické struktury pro lokalizaci bodů v rovině; binomiální a Fibonacciho haldy, leftist haldy a 2-3 haldy. Též několik přátelských randomizovaných datových struktur: skip listy a treapy.

Grafy & algoritmy II

GA2

Martin Böhm, Karel Tesař, Pavel Veselý

Pokročilejší grafové algoritmy: union-find problem, hledání minimální kostry, párování v grafech, testování vícenásobné souvislosti a silné souvislosti.

Intervalové stromy * (“Já bych ty intervaly nejradši. . . dal do stromu!”)

ITREE

Karel Tesař

Intervalový strom je datová struktura pracující s intervaly, se kterou se můžeme setkat v mnoha úlohách (zejména soutěžních). Řekneme si, co to intervalový strom je, jaké všechny druhy intervalových stromů existují a jejich použití si ukážeme na úlohách. Na závěr si představíme jednu „magickou“ datovou strukturu jménem Fenwickův strom.

Toky v sítích * (“Když je v grafu povodeň, těsní?”)

TOKY

Martin Böhm, Karel Tesař, Pavel Veselý, Jirka Setnička

K čemu je dobré, když grafem teče voda. Předvedeme si klasický problém toků v sítích a jeho všelijaké, mnohdy dosti překvapivé aplikace. Jak rozestavět n věží na šachovnici a jak ji místo toho pokrýt dominovými kostkami? Další souvislosti, jako třeba násobná souvislost grafů.

Předpoklady: Umět plavat (zejména v matematice)

Hledání v textu (“»Vyšíváme v seníku!« – kde jsem to jen viděl?”)

REGEX

Karel Tesař, Jirka Setnička

Vyhledávání čehokoliv ve velkém množství textu: Prostá vylepšení hledání hrubou silou: Karp-Rabin, Boyer-Moore. A algoritmy chytřejší: Morris-Pratt, Knuth-Morris-Pratt, Aho-McCorasicková. Konečné automaty prakticky, regulární a „regulární“ výrazy.

Geometrie a počítače (“Nerušte mé kruhy! (ani jiné kvadriky)”)

GEOM

Martin Mareš, Jirka Setnička

Základní algoritmy pro řešení geometrických úloh – konvexní obal, dva nejbližší body v rovině, výpočet obsahu nekonvexního mnohoúhelníka, lokalizace bodu, scanline algoritmus a jeho použití, Voroného diagramy a souvislost s persistentními datovými strukturami.

Herní algoritmy (“Když nemáte na to, abyste vyhráli šachový turnaj. . .”)

AIGAME

Pavel Veselý

Povídání o tom, jak programovat počítačové soupeře do šachů a her jim podobným. Základní minimaxový algoritmus a jeho vylepšení neboli α - β ořezávání. Stále pomalé? Několik nápadů na efektivnější ořezávání. Ne u všech her však funguje hrubá síla (minimax) dobře, ukážeme si tedy, jak hru zanalyzovat.

Logické úlohy a informatika

LOGINF

Karel Tesař

Povíme si, jak logické úlohy souvisí se základy informatiky. Ačkoliv to tak na první pohled nevypadá, tak ve většině logických úlohách jsou skryty informatické postupy jako je například kódování informací, hledání správné cesty nebo splňování podmínek.

Programovací jazyky a techniky

Programování v jazyce C

C

Pavel Veselý, Martin Böhm, Radim „Rumcajz“ Cajzl, Jirka Setnička

Datové typy jazyka C, programové konstrukce, základy práce s ukazateli. Seznámení se standardními knihovnamí jazyka C.

Programování v jazyce C# (“Co se stane, když strčíte Céčko za mřížku?”)

CIS

Pavel Veselý, Tomáš Maleček

C# je moderní objektově orientovaný jazyk, který za víc než deset let svého bouřlivého vývoje dostal do vínku některé funkcionální rysy. Mimo popisu základních konstrukcí si projdeme zajímavé části dotnetí Base Class Library.

Objektově orientované programování v C++ a Object Pascalu (“I život je objektový, tak proč ne programování. . .”)

OBJ

Honza Škoda

Objektově orientované programování přináší jiný náhled na návrh řešení problémů. Vysvětlíme, jak se liší objektové a procedurální programování. Co je to objekt a co třída. Základní vlastnosti objektů (dědičnost, zabalení, polymorfismus). Co je to metoda, překrývání metod, virtuální metody (pozdní vazba) a čistě virtuální (abstraktní) metody. Syntaxe deklarací tříd v C++ a Delphách, viditelnost a přístupová práva ke členům tříd. A také, že to není jediný přístup k objektům.

Předpoklady: Znalosti procedurálního programování v Pascalu a/nebo v C.

Základy webu (“Proč se cpát do Flashe, když to jde hezky a standardně jinak?”)

WWW

Tomáš Maleček, Honza Škoda, Jirka Setnička

Úvod do servírování informací skrze World Wide Web. Co znamená `http://`, jaký je rozdíl mezi HTML a XHTML a na co se hodí znát kaskádové styly. Kdy a proč (ne)použít kterou technologii.

Tomáš Maleček, Honza Škoda, Jirka Setnička

Úvod do problematiky vytváření dynamicky generovaných webových stránek. Syntaxe jazyka PHP, netypovost, (ne)pole. Jak psát stránky, za které se nemusím stydět, objektové programování, pár slov o bezpečnosti. Zamícháno s XHTML, JavaScriptem, CSS a možná i (My)SQL.

Předpoklady: WWW

Jazyk XML a související technologie (“<xml style="vesele"/>”)

Tomáš Maleček

Povíme si, co je to jazyk XML, jak vznikl a k čemu je dobrý. Zároveň probereme nástroje pro práci s XML dokumenty, jejich načítání (parsování), generování, validaci atd. Z pokročilejších technik pak ukážeme vyhledávání v XML pomocí jazyka XPath a transformace XML dokumentů pomocí XSLT. Zbyde-li čas ukážeme si i několik konkrétních aplikací (XHTML, WML, SVG, ODF, ...).

Haskell * (“fac n = product [1..n]”)

Martin Böhm

Funkce v obvyklých (imperativních) programovacích jazycích nejsou funkce v matematickém slova smyslu: mají „vedlejší efekty“, projevují se na běhu programu i jinak, než svou návratovou hodnotou. Ve funkcionálních jazycích, mezi něž Haskell patří, mají takové chování zakázáno. K čemu to?

Programování v UNIXu (“Segmentation fault. Core dumped.”)

Martin Mareš

Jak se programuje v operačním systému, který si programátoři navrhli pro sebe? Vyměníme klikací vývojová prostředí za stavebnici se spoustou kostiček. Textový editor Vim, překladač GCC a otrokář zvaný make. Jak zjistit, co program doopravdy dělá, i když běží hodinu – core dumpy, GDB, valgrind, oprofile. Jak programovat ve více lidech a nezbláznit se z toho – diff, patch, Git.

Prolog * (“Co s jazykem bez přiřazovacího příkazu?”)

Pavel Veselý, Tomáš Maleček

Proč psát dlouhé a složité programy, když stačí dostatečně přesně popsat situaci a pak se prostě zeptat? Toť princip logického programování, který si ukážeme na Prologu.

Python (“import antigravity”)

Honza Škoda

Základy rychlého programování v Pythonu. Syntaxe, datové typy, funkce, třídy, ... Na co si dát pozor, v čem se liší od ostatních jazyků a proč je tak oblíbený.

Matematické přednášky

Diskrétní matematika (“O Dlouhém, Širokém a šatnářce”)

Pavel Veselý, Karel Tesař

Úvodní minikurs diskretní matematiky (to je opak matematiky spojitě, čili mimo jiné kombinatorika). Seznámení s relacemi a jejich vlastnostmi. Dozvíte se také něco o uspořádaných, nezávislých a jiných množinách. S pomocí kombinatoriky možná vyřešíme problém zmatené šatnářky. Hallova věta nám pomůže určit, jestli má cenu snít o perfektním párování.

Derivace a integrály pro zelenáče

Radim „Rumcajz“ Cajzl

Rychlokurz derivování a integrování. Rychle prolétneme limity, nadefinujeme si derivace a procvičíme jejich výpočty. Dále si řekneme, co je to integrál, jak se definuje a počítá. Na závěr si ukážeme, k čemu je to všechno dobré v reálném či středoškolském světě – „rychlé“ odvozování fyzikálních vzorců, grafy funkcí, všemožné optimalizace.

Komplexní čísla (“ $1 = \sqrt{-1}^2 = (\sqrt{-1})^2 = i^2 = -1$. WTF???”)

Radim „Rumcajz“ Cajzl

Ukážeme si, co se stane, když začneme počítat s čísly mimo reálnou osu: co je to imaginární jednotka a jak se s ní počítá, komplexní čísla a počítání s nimi v algebraickém, goniometrickém a exponenciálním tvaru, dále převod mezi těmito tvary. Zobrazení a výpočty v Gaussově rovině. Také si řekneme, k čemu se komplexní čísla dají použít v praxi.

Lineární algebra (“Vektorový prostor je místo, kde žijí vektory.”)

Martin Böhm, Radim „Rumcajz“ Cajzl, Pavel Veselý

Lineární algebra vznikla jako formalizace geometrie a tuto souvislost si ukážeme. Popíšeme vektorové prostory, které se skládají z vektorů. Jaké operace s nimi umíme provádět a co všechno musí splňovat? Kdy jsou vektory závislé a kdy nezávislé? Co je to lineární kombinace, obal a generátor? Co je to dimenze vektorového prostoru a jaké má souvislosti s předchozími pojmy?

Kryptologie (“Gbgg arav zbp gnwan mcenin.”)

Martin Mareš, Jirka Setnička

Kryptologie čili tajuplná nauka o šifrách, jejich konstrukci a hlavně o jejich luštění. Přísně tajné. Šifrovací systémy jako lego: základními kostičkami nám budou symetrické a asymetrické šifry a jednosměrné funkce, stavět z nich budeme kryptografické protokoly na bezpečný přenos, autentikaci, digitální podpisy a třeba i jak si hodit korunou po telefonu. Předvedeme nerozluštitelnou šifru a dokonce to o ní i dokážeme.

Teorie kombinatorických her (*“Život je jen hra. . . Jakou má vyhrávající strategii?”*)

GAME

Pavel Veselý

Rozličné kombinatorické hry se zápalkami, kamínky, barvičkami či grafy a jejich abstrakce. U některých si ukážeme výherní či obranné strategie, u některých dokážeme, že sice příslušná strategie existuje, ale nikdo ji ve skutečnosti nezná. Teorie pana Conwaye: jak pozicím přiřazovat čísla, popř. jiné symboly. Piškvorky a jejich zobecnění na hypergrafy.

Teorie grafů

GRAF

Martin Böhm, Martin Mareš

Grafy, které vás na střední škole neučili! Žádné osy a funkce, jsou to města a silnice. Jednoduché, představitelné objekty. Často je vidáme v algoritmech, ale i z čistě matematického úhlu pohledu jsou zajímavé. Ukážeme si nečekaná použití grafů v geometrii, teorii čísel i jinde. Na grafech si také představíme dualitu – vlastnost, kterou mají snad všechny matematické pojmy společnou. Vše jednoduše a srozumitelně i pro ty, kdo o grafech slyší prvně.

Rovinné grafy

ROG

Pavel Veselý, Karel Tesař, Jirka Setnička

Povídání o grafech, které jde nakreslit na papír bez křížení hran. O tom, co všechno pro takové grafy platí a jak je poznáme, aniž bychom je museli kreslit. Existuje pouze 5 pravidelných mnohostěnů a my se o tom pomocí teorie grafů přesvědčíme. Barvení rovinného grafu šesti a možná i méně barvami. Když zbude čas, zkusíme grafy kreslit i na jiné plochy: kupříkladu Möbiovu pásku, pneumatiku nebo ušatou kouli.

Hardware, operační systém a další technické

UNIX (*“UNIX gives you enough rope to hang yourself.”*)

UNIX

Honza Škoda, Martin Mareš

Kamarád u černobílého textového okna září blahem. Chcete poznat, proč? Jak UNIX vznikl, k čemu je dobrý a k čemu třeba není. UNIXová filosofie. Kouzlo scriptů. Kouzlo speciálních souborů. Kouzlo propojování programů. Kouzlo nechtěného. UNIX byl napsán v C a C vzniklo pod UNIXem.

Principy počítačů (*“A opravdu uvnitř počítače běhají malí trpaslíci?”*)

HW

Martin Mareš, Tomáš Maleček, Jirka Setnička

Historie a vývoj počítačových technologií a současných architektur. Co je to procesor, jak se programuje (instrukční sady) a jak se chová. Přehled tříd procesorů (skalární a vektorové, RISC). Jak procesor komunikuje s okolím (čipová sada). Nezbytné periferie: vnitřní a vnější paměti, vstupní a výstupní zařízení.

TEX

TEX

Martin Mareš, Radim „Rumcajz“ Cajzl

Donald E. Knuth napsal TEX před desítkami let proto, že mu nikdo nebyl schopn vysázet matematický text podle jeho požadavků. Od té doby se hojně používá pro sazbu nejrůznějších publikací. Probereme základy TEXu, jeho využití při psaní řešení KSP, ale třeba také povídek, ročenky, slajdů pro prezentaci, zpěvníku nebo rovnou not.

Hradla (*“Vše co jste chtěli vědět o hradlech a báli jste se zeptat.”*)

HRAD

Radim „Rumcajz“ Cajzl, Jirka Setnička

Shrnutí hradlového seriálu KSP. Tedy krátký úvod do historie, nástin vnitřností, schémata, schematické značky, konvence. Trocha teorie, Booleova algebra a De Morganovy zákony. V neposlední řadě si ukážeme, jak rychle a efektivně umíme problémy hradly řešit.

Počítačová grafika (*“Namaluj mi beránka. . .”*)

GFX

Pavel Veselý, Martin Mareš

Kreslení a zpracování obrazu na počítači. Souřadnice (rovinné, prostorové i barevné) a jejich transformace. Základní grafická primitiva: body, úsečky, kružnice, elipsy, Bézierovy křivky a jejich rasterizace. Vyplňování n -úhelníků a křivkou ohraničených oblastí, flood fill. Pár triků navíc: maticové filtry, anti-aliasing a dithering. Grafické formáty a komprese obrázků. Základy trojrozměrného promítání a vykreslování scény.

Tabulkový procesor z příkazové řádky (*“Když vás začne Excel štvať. . .”*)

CALC

Radim „Rumcajz“ Cajzl

Zpracování menšího množství dat – máte tabulku dat uloženou v textovém souboru a chcete k ní přidat několik dalších sloupců. Ukážeme si polopatický způsob pomocí libovolného programu, dále zrychlení přes program calc (neplést s oocalcem). Dále si ukážeme, jak z takové tabulky nakreslit graf pomocí programu GNUplot, včetně pokročilejších funkcí – fitování dat funkcí, více grafů v jednom, přepočítání dat za běhu, . . . Také si ukážeme integraci s TEXem.

Sítě a Internet (*“Sítě nejen na ryby.”*)

NET

Martin Mareš

Jak funguje Internet a počítačové sítě vůbec. Lokální sítě s dráty i bez nich a různé způsoby, jak je mezi sebou propojovat. Protokoly rodiny TCP/IP a nad nimi postavené aplikační protokoly: DNS, SMTP, HTTP a celý zvěřinec dalších. Bezpečnost sítí a všelijaké útoky na ni. Pár taktů hudby budoucnosti: IPv6, multicasting, přenos v reálném čase atd.

Ostatní

Úvod do programátorských soutěží (*“Za pár let chci třímát v rukou pohár!”*)

SOUT

Karel Tesař, Martin Böhm

Tipy a triky, jak uspět v programátorských soutěžích a olympiádě. Čeho si všimnout při vymýšlení algoritmů a na co si dávat pozor při samotné implementaci. Jakých soutěží se na střední škole můžete účastnit, kde se dají získat zkušenosti a kde se naopak dají vyhrát velké ceny.

Půlnoční přednášky

MFF UK aneb co obnáší matfyzákem býti (*“Mamínko, ptá se tatínka, kdy už budu matfyzákem?”*)

MFF

Nezávazné povídání o Matfyzu a základním matfyzáckém folklóru. Určitě si přečteme matfyzáky sepsané Úvod do matfyzáka a zazpíváme pár matfyzáckých písní. Zbytek už bude záležet na tom, co budete chtít slyšet.

Matfyzácké vtipy (*“Proč nemůže být epsilon záporné?”*)

FTIP

Karel Tesař

Vše o matfyzáckých vtipech a jejich rozdělení do kategorií. Které z nich jsou ty pravé matfyzácké? U kterých zas stačí jen dosadit jméno své školy a pořad zůstanou vtipné? A které vtipy pochází od normálních lidí a snaží se akorát matfyzáky pomlouvat?

Mapování mysli a jiné techniky (*“Někdo mapuje terén, my mapujeme mysl!”*)

MIND

Pavel Veselý

Mapy mysli (neboli myšlenkové mapy) jsou speciální diagramy, které mají v člověku povzbudit kreativitu. Kromě vymýšlení všemožných věcí od přednášky po výpravu na Mars je lze využít pro organizaci myšlenek a při učení. Na příkladě si tedy nějakou mapu mysli vytvoříme a představíme si program na jejich tvorbu. V případě času rovněž: brainstorming, duševní rozcvičky, nasazování klobouků a další užitečné kreativní techniky.

Ďábelské herní algoritmy * (*“Vyždímáme z počítače veškerou jeho sílu za pomoci chytřejších algoritmů?”*)

AIGAME2

Pavel Veselý

Minimax s α - β ořezáváním nestačí? Pokročilejší techniky na efektivnější ořezávání. Jiné přístupy než minimaxovaný algoritmus aneb Proof number search a metody Monte Carlo.

Předpoklady: AIGAME

Klávesnice (*“Ono existuje ještě další vstupní zařízení?”*)

KEYB

Martin Böhm

Vnitřek klávesnice je jedno z nejspinavějších míst ve vašem domě. Povíme si, co se v ní dále nachází. Někteří lidé si nemohou vynachválit klávesnice mechanické, jiné ergonomické – a jiné ty zcela normální. Budeme hovořit o vlastnostech všech tří. Slovem zavedeme i o rozložení kláves a jejich (údajných) vlivech na rychlost psaní. To vše s příklady na vlastní prsty!

Základy první pomoci (*“Jak někomu zachránit život a jak málo k tomu stačí?”*)

ZDRAV

Jirka Setnička

Pobavíme se o základech první pomoci. Jak správně vyhodnotit situaci a kdy je potřeba volat pomoc? Jak se postarat o člověka v bezvědomí, jak kontrolovat životní funkce a jak člověka stabilizovat do příjezdu pomoci? Ukážeme si, jak málo stačí k záchraně života a naučíme se nebát se první pomoci. A také, že naše bezpečí je v každé situaci na prvním místě.

Populární pyrotechnika pro děti (*“Celé opdoledne si zkouším různé směsi, ale ono to prostě nebouchá!”*)

BUM

Radim „Rumcajz“ Cajzl

Povídání o tom, jak se podomácku vyrábějí výbušniny, na co se přitom musí dávat pozor a co je ještě bezpečné. Popíšeme vlastnosti a výrobu střelného prachu, různých flashových směsí, amonledkových trhavin, jododusíku a některých nitrosloučenin. Dále něco málo o způsobech odpalu, proč je zápalná šňůra nebezpečná. Doplněno o spoustu historek z vlastní zkušenosti. Bude-li zájem, vysvětlíme si fyzikálně-chemickou podstatu výbuchu a jak se spočítá energie při výbuchu. Některé s probraných látek si ukážeme v praxi.

Teorie množin * (*“Je Vlk nedosažitelný kardinál?”*)

TEMNO

Martin Mareš

Teorie množin tvoří páteř veškeré matematiky. Pomocí množin se totiž modelují veškeré objekty, které se v matematice vyskytují. Celou teorii prostupuje magický pojem *nekonečno*. Jakým způsobem se tohoto, pro spekulativní mysl ošidného, termínu zhostila moderní matematika? Množiny a jejich velikosti. Cantorův diagonální trik. Ordinaly a houšť kardinálů. Potenciální kontra aktuální nekonečno. Myslíte si, že máte dobrou představu o tom, co jsou přirozená čísla? Možná vás z ní vyvedeme. A co teprve reálná čísla. Dva zneprátené axiomy.