Jarní soustředění KSP 2014 – 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 si o to zkuste říct, třeba se najde někdo, kdo by vám o tom rád pověděl.

Přednášky jsou členěné do třech kategorií. Na ty, o kterých si myslíme, že jsou základní a esenciální pro každého programátora, na přednášky pokročilejší, dotýkající se zajímavých oblastí informatiky, a nakonec na přednášky půlnoční, které často rozšiřují obzory až za rámec informatiky. 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 dæmonologie 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. Hvězdičky znamenají obtížnost.

Základní přednášky

Mezi těmito přednáškami jsou věci, které by měl každý začínající programátor umět. Bez pochopení většiny věcí přednášených na těchto přednáškách se budete na pokročilých přednáškách, které na ně navazují, jen obtížně chytat. Doporučujeme proto nejdříve zvládnout tyto přednášky a osvěžit si nějaký základní programovací jazyk, než se pustíte do pokročilejších věcí.

Základy programování *("Má $x = x + 1$ řešení?")* **ZAKL**
Karolína Burešová, Jan Škoda
Úvodní několikadílná přednáška pro ty, kteří mají s programováním jen malé, nebo dokonce žádné zkušenosti. Od základů si vysvětlíme problematiku programování. Pracovat budeme v jazyce Python, který je jednoduchý na naučení, a přesto zároveň velmi mocný. Ukážeme si základní datové typy (n-tice, seznamy, slovníky), datové struktury (fronta, zásobník) i několik algoritmů a programátorských technik, bez kterých se žádný programátor neobejde.

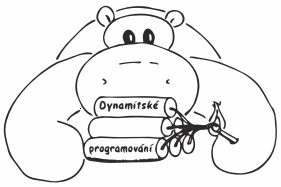
Základy algoritmizace, složitosti a datových struktur *("Co by měl každý prográmátor znát.")* **ZALG**
Karel Tesař, Martin Španěl
Základní kurz algoritmů a datových struktur. Jak poznat který algoritus je efektivnější? Přehled základních algoritmů. Co je to datová struktura a několik jejích ukázek. Vše si procvičíme na příkladech.

Grafy & algoritmy I *("Pokud jste ještě neslyšeli o informatických grafech, tak tato přednáška je právě pro vás.")* **GA1**
Karel Tesař, Lukáš Folwarczný
Úvod z teorie grafů a použití grafů při řešení algoritmických problémů. Naučíme se hledat nejkratší cestu v bludišti, pochopíme základní princip, jak funguje GPS navigace a mnohé další. Vše si procvičíme na konkrétních příkladech.

Dynamické programování *("Kampak jsem si to jenom schoval?")* **DYNP**
Karolína Burešová, Karel Tesař
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.

Základy teorie čísel a geometrie v algoritmech **TEGE**
Karel Tesař, (Jirka Setnička)
Jak spočítat obsah mnohoúhelníku? Co je to konvexní obal? Jak se efektivně počítá k-tá mocnina čísla n? Jak efektivně v úlohách počítat modulo prvočíslo? A mnohé další.

Hledání v textu *("»Vyšíváme v seníku!« – kde jsem to jen viděl?")* **TEXT**
Karolína Burešová, Martin Španěl, Jirka Setnička
Někdy potřebujeme najít podřetězec ve velkém množství textu. Stromeček trochu připomínající ten biologický aneb trie. Proč se ve vstupu vracet neboli Knuthův-Morrisův-Prattův algoritmus. Hledání více řetězců najednou podle Aha a Corasickové. Okénkové hešování Rabina a Karpa.



Základní programovací jazyky a techniky

Programování v jazyce C **C**
Karolína Burešová, Michal Pokorný, Karel Tesař, Jirka Setnička, Jenda Hadrava
Jazyk C patří k nejrozšířenějším jazykům, hodí se pro low-level programování i kusy kódu, které mají zejména být rychlé. Představíme si datové typy a běžné programové konstrukce, vysvětlíme si základy práce s ukazateli a také se seznámíme se standardními knihovnami jazyka C.

Python **PYTH**
Karolína Burešová, Jirka Setnička
Jak programovat v Pythonu a jak v něm „nepsat Cčko“. Syntaxe, datové typy, funkce, třídy, ... Na co si dát pozor, v čem se Python liší od ostatních jazyků a proč je mezi nimi tak oblíbený.

C++ a objektově orientované programování *(“I život je objektový, tak proč ne programování...”)* **OOP**
Michal Pokorný, Jan Škoda
Každý odrostlý programátor musí umět programovat objektově! Vysvětlíme si, co to znamená a čemu tím pomůžeme, co je to objekt a co třída. Ukážeme si základní vlastnosti objektů (hlavně dědičnost a zabalení) a různé další speciality jazyků s OOP, například virtuální a abstraktní metody, pozdní vazbu a polymorfismus nebo překrývání metod. Dá-li čas, možná se stihne i pár návrhových vzorů. Vše budeme předvádět na C++.
Předpoklady: Znalost programování, nejlíp v něčem podobném C.

Pokročilé přednášky

Tyto přednášky by měly jednak dále rozvíjet znalosti ze základních přednášek, ale také nabízet další zajímavé programátorské techniky a technologie, které se mohou každodenně hodit.

Algoritmizace

Intervalové stromy * *(“Já bych ty intervaly nejradši... dal do stromu!”)* **ITREE**
Karolína Burešová, 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 Španěl, Jirka Setnička, Jenda Hadrava
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.

Datové struktury pro pokročilé * *(“Pojďme na procházku binárním lesem”)* **DS**
Martin Mareš, (Karel Tesař)
Přehled šikovných datových struktur, které se nevešly do ZALG. Vyhledávací stromy a různé způsoby jejich vyvažování a „ozdobení“. Hešování aneb hledáme v téměř konstantním čase. Líné datové struktury a amortizovaná složitost.

Geometrie a počítače *(“Nerušte mé kruhy! (ani jiné kvadriky)”)* **GEOM**
Jirka Setnička, Martin Mareš
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**
Karolína Burešová, Martin Španěl
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.

Parsing čili analýza textu *(“1+2*4 = 12”)* **PARSE**
Martin Mareš, Jirka Setnička
Často potřebujeme načíst nějaký složitý textový vstup: matematický výraz, webovou stránku v HTML, zdroják programu, ... Ukážeme si, jak texty analyzovat (neboli parsovat), aniž bychom v nich zabloudili: rozdělení na lexikální a syntaktickou vrstvu, železničářský algoritmus na parsování výrazů, popis syntaxe pomocí regulárních výrazů a gramatik.

Těžké problémy * **HARD**

Lukáš Folwarczný, Martin Mareš

V rámci této přednášky se budeme zabývat problémy tak těžkými, že nikdo na světě pro ně neumí vymyslet efektivní (rozuměj polynomiální) algoritmus. Spousta lidí dokonce věří, že to vůbec možné není. Abychom mezi tyto problémy pronikli, seznámíme se s pojmy NP-úplnosti a NP-těžkosti. Především si však konkrétní těžké úlohy ukážeme a naučíme se i některé těžké úlohy rozpoznat. Závěrem si řekneme, jak se s těžkými úlohami vypořádat v praxi.

Další programovací jazyky a techniky

C++ * **CPP**

Michal Pokorný, Jan Škoda

Pro znalé základů nabízíme středně hlubší seznámení s C++. Určitě přijdou na řadu reference, přetěžování operátorů, metaprogramování pomocí templatů a mechanismus výjimek. Pokračovat se bude běžně používanými funkcemi standardní knihovny jazyka, jako jsou například I/O streamy nebo užitečné datové struktury a algoritmy. Nakonec můžeme za odměnu ukázat některé novinky ze standardu C++11.

Předpoklady: C, OOP

Programování v jazyce C# *("Co se stane, když strčíme Céčko za mříže?")* **CIS**

Michal Pokorný

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.

Programování v jazyce Java **JAVA**

Karel Tesař

Java je jeden z nejrozšířenějších objektových programovacích jazyků za posledních deset let. Na přednášce se seznámíme s jeho myšlenkou a naučíme základy. Přednáška je dělaná pro posluchače, kteří umí alespoň základy jiného programovacího jazyka.

Ruby * *("Když to plave a kváká, je to kachna.")* **RUBY**

Michal Pokorný

Ruby je vhodný jak na jednoúčelové skriptíky, tak na megaweby se spoustou megapřístupů denně. Je objektově orientovaný, ale nehází klacky pod nohy moc silným typováním. Syntaxi má snadno čitelnou a elegantní, avšak (téměř) bez speciálních případů. Ukážeme si letem světem základy jazyka a pak si jako motivační příklad nakódíme web ve frameworku Rails. V případě zájmu může přednáška pokračovat druhou částí.

Předpoklady: Nejmíň jeden objektově orientovaný jazyk.

Perl *("Jak Pejsek a Kočička vymýšleli programovací jazyk")* **PERL**

Karolína Burešová, Martin Mareš, Jirka Setnička

Jednoho dne se Larry Wall rozhodl, že nasype do jednoho velkého kotle spousty programovacích jazyků a unixových utilit, za stálého míchání povaří, posléze přecedí, přikoření a implementuje. Tak vznikl Perl, jazyk původně určený hlavně na zpracování textu, ovšem jak se ukázalo, též šikovný na spoustu dalších věcí. Asociativní pole, libovolně složité datové struktury za pomoci referencí, balíčky a objekty zdarma a hlavně regulární výrazy zde a všude. Zkrátka jazyk, který lze jedině milovat nebo nenávidět, nic mezi tím.

Prolog * *("Co s jazykem bez přiřazovacího příkazu?")* **LOGP**

Karolína Burešová, Jirka Setnička

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.

Webové stránky **WWW**

Karolína Burešová, Martin Mareš, Jirka Setnička

Co se děje za oponou, když do prohlížeče zadáte adresu svých oblíbených stránek? A jak si takovou stránku taky pořídit? Přelet nad protokolem HTTP, seznámení s HTML a předvedení kaskádových stylů. Jak fungují dynamické stránky od formulářů až po JavaScript běžící v prohlížeči.

Jazyk SQL *("SELECT something FROM knowledge LIMIT 90min")* **SQL**

Karolína Burešová, Martin Mareš, Jirka Setnička

Představíme si SQL, jazyk databází. Ukážeme si základní příkazy i práci o kus složitější. Jak ušetřit skriptu práci a sobě čas, aneb jak se zeptat rovnou na to, co chci vědět. K čemu se hodí složený dotaz a klíčové slovo `JOIN`.

Počítačová grafika *("Namaluj mi beránka...")* **GFX**

Jirka Setnička, Martin Mareš

Kreslení a zpracování obrazu na počítači. Co vše obnáší vykreslení obyčejné čáry, aby to bylo rychlé a pěkně vypadalo. A co teprve, když ta čáry zatáčí! Vyplňování n-úhelníků a křivkou ohraničených oblastí, flood fill. Také maticové filtry pro zpracování fotek (zaostření, rozmazání), anti-aliasing a dithering. Pokud se stihne, tak navíc základy 3D vykreslování.

Hardware, operační systémy a další technikálie

Principy počítačů *("A opravdu uvnitř počítače běhají malí trpaslíci?")* **HW**
Martin Mareš, Jirka Setnička, Jenda Hadrava
Vydáme se do země skřítků, kteří pohánějí počítače. Počítačové architektury od hodinek po superpočítač od Craye, jejich křivolaká historie i současnost. Co je to procesor, jak se programuje a jak se chová. Různé druhy pamětí a jejich cacheování. Jak procesory komunikují s okolím – sběrnice, čipové sady, vstupní a výstupní zařízení. A co když je procesorů několik, nebo třeba pár tisíc? Přednáška bude praktická: pár počítačů při ní rozebereme a možná i nějaký postavíme.

Hradla *("(A `NOR` B) `NAND` (A `XOR` B) = 1")* **HRAD**
Jenda Hadrava
Krátký úvod do historie, nástin vnitřností, schémata, schematické značky, konvence. Povíme si něco z teorie, o Booleově algebře a De Morganových zákonech. Podíváme se na spojitost s digitálními obvody. V neposlední řadě si ukážeme, jak rychle a efektivně umíme problémy hradly řešit.

Od zdrojáku k programu *("Před spuštěním program přeložte. Stačí třikrát podélně?")* **KOMP**
Martin Mareš
Mezi programem v Céčku, který jste právě dopsali, a tranzistory uvnitř vašeho procesoru leží obrovské území obývané překladači, linkery, knihovníky, operačními systémy, loadery a jinými bájnými bytostmi. Pojďme zjistit, co jsou zač a co všechno s programem provádějí. Co udělá kompilátor za nás a co musíme naopak udělat my za něj.

UNIX *("UNIX gives you enough rope to hang yourself.")* **UNIX**
Martin Mareš, Jan Škoda, Jenda Hadrava
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 skriptů. Kouzlo speciálních souborů. Kouzlo propojování programů. Kouzlo nechtěného. UNIX byl napsán v C a C vzniklo pod UNIXem.

Sítě a Internet *("Sítě nejen na ryby.")* **NET**
Martin Mareš, Jirka Setnička, Jenda Hadrava
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.

Systém pro správu verzí Git *("U svatýho tučňáka, kdo sem napsal tohle? Ono to tvrdí, že JÁ?!")* **GIT**
Karolína Burešová, Michal Pokorný, Martin Mareš, Jirka Setnička, Jenda Hadrava
Když se něco vyvíjí delší dobu, přijde vhod nějaký sofistikovaný nástroj. Na jeden takový, Git, se podíváme. Povíme si, jak Git ukládá změny, co jsou commity, větve, tagy a jak vypadá merge mezi větvemi. Nakonec možná předvedeme i nějaké kouzelnické triky: hledání bugů půlením historie, přepisování dějin.

Aplikace informatiky

Počítačová lingvistika *("Jsou bramborové knedlíky plněné bramborami?")* **CMPLING**
Karolína Burešová
Zejména motivační přednáška o počítačové lingvistice a počítačovém zpracování přirozeného jazyka. Podíváme se na vlastnosti přirozených jazyků a zaměříme se na to, jak moc komplikují jejich počítačové zpracování. Pojmenujeme odlišnosti mezi kontrolou pravopisu, automatickým překladem a konverzací s uživatelem a ukážeme si, co se zatím umí používat.

Testování uživatelského rozhraní *("Vždyť to tlačítko je tak evidentní!")* **TUR**
Karolína Burešová
Obvykle tvoříme programy (nebo třeba webové stránky) s cílem, aby je používali i další lidé. K tomu je ovšem vhodné, aby se i ostatním dobře používali. Jak něco takového měřit a testovat? Kognitivní průchod, heuristická evaluace i testování s lidmi. Jak pracovat s výsledky testů a proč nevadí, že Vim by v některých testech rozhodně neuspěl.

TEX *("No pages of output. Ask a TEXnician.")* **TEX**
Karolína Burešová, Martin Mareš, Jirka Setnička
Donald E. Knuth napsal TEX před desítkami let proto, že mu nikdo nebyl schopen vysázet matematický text podle jeho požadavků. Od té doby se hojně používá pro sazbu nejrůznějších publikací. V této spíše praktické přednášce si ukážeme použití TEXu od hladké sazby knihy až po zběsilosti hraničící s programováním. Pozornost věnujeme i zdrojům informací a rozdílům mezi různými dialekty TEXu.

Komprese dat *("Jnm idln kpln j nstlčtln.")* **ZIP**
Martin Mareš
Přehled základních kompresních algoritmů: triviální algoritmy (RLE), statistické metody (Huffmanovo a aritmetické kódování), slovníková komprese (LZ77, LZ78, LZW), Burrowsova-Wheelerova transformace (BZIP). Pokud zbude čas, tak i něco o ztrátové kompresi obrázků a zvuku (prediktory, wavelets, JPEG, MPEG, fraktály).

Čárové kódy *("Jak naučit počítače číst láhve od Coly")* **BAR**
Martin Mareš
Čárové kódy dnes potkáváme na každém kroku, ale jak doopravdy fungují? Prozkoumáme klasické jednorozměrné kódy (UPC, EAN, Code39, Code128), jakož i novější dvojrozměrné (QR, Aztec, DataMatrix). Kódovací a dekódovací algoritmy plus trocha matematiky okolo zabezpečení proti chybám. Další počítačem čitelné značky: RFID, bíle křížky na asfaltu, ...

Matematické přednášky

Matematika náhody *("Světem vládne náhoda a současně i s ní zákon, který vzniká z obrovského souboru náhod.")* **PST**
Lukáš Folwarczný
Prozkoumáme, jakým způsobem matematika zachází s pojmem náhody. Podíváme se na motivaci a způsob zavedení teorie pravděpodobnosti, odvodíme pár užitečných vztahů a především budeme řešit zajímavé příklady. Pokud budete všemi pravděpodobnostními příklady zmoženi, přečteme si část Dopisů o pravděpodobnosti.

Kryptologie *("Gbgb arav zbp gnwan mcenin.")* **CRYPT**
Karolína Burešová, Martin Mareš, Jirka Setnička, Jenda Hadrava
Kryptologie se zabývá šiframi, jejich konstrukcí a zejména jejich luštěním. Začneme se symetrickými a asymetrickými šiframi a jednosměrnými funkcemi. Z nich pak vybudujeme složitější kryptografické protokoly na bezpečný přenos, autentikaci a digitální podpisy. Vymyslíme dokonce, jak si hodit korunou po telefonu, a také předvedeme nerozluštitelnou šifru.

Diskrétní matematika *("O Dlouhém, Širokém a šatnářce")* **DM1**
Martin Španěl, Karel Tesař, Martin Mareš, Lukáš Folwarczný
Úvodní minikurz diskrétní matematiky (to je opak matematiky spojité, 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.

Teorie nemožného * *("Neexistence důkazu není důkazem neexistence. Dokažte.")* **NONEX**
Martin Mareš
Existenci slona v Africe snadno dokážete tím, že ho přivedete. Jak ale ukázat, že tam žádný slon není, případně že sice je, jenže ho nejde najít pomocí pravítka, kružítka a jeepu? Přímo se to dělá těžko, ale existuje spousta krásných triků, jak neřešitelnost problémů dokazovat. Nesložitelné hlavolamy, nerozvázatelné uzly, nepopsatelná čísla, neroztřetitelné úhly, nealgoritmické problémy a jiné slasti nekonstruktivní matematiky. Jak naopak ukázat, že něco existuje, aniž bychom věděli, jak to vypadá?

Teorie metrických prostorů * *("Jak se vzdálenost měří na Manhattanu a jak v Paříži?")* **TMP**
Lukáš Folwarczný
Metrický prostor je matematická struktura zobecňující různé způsoby měření vzdálenost. Představíme si některé známé užitečné příklady metrik, dokážeme si pár základních obecných vlastností metrických prostorů a procvičíme si představivost. Přednáška proběhne v rychlejším tempu a může se hodit letmá znalost limit posloupností.

Derivace a integrály **DIFF**
Jenda Hadrava
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. Hlavně si však ukážeme, k čemu je to všechno dobré v reálném či středoškolském světě – „rychlé“ odvozování fyzikálních vzorců, grafy funkcí, všemožné optimalizace.

Půlnoční přednášky

Aneb přednášky přednášené (nejen) o půlnoci na různá zajímavá témata nejen o informatice. Pokud nějaká z nich nebude oficiálně vypsaná, je možné si konkrétního organizátora ve volné chvíli chytit a přesvědčit ho k přednášení.

Úvod do programátorských soutěží *("Za pár let chci třímat v rukou pohár!")* **SOUT**
Karel Tesař & Lukáš Folwarczný
Tipy a triky, jak uspět v programátorských soutěžích a olympiádě. Čeho si všímat 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.

Programování pro Android **ANDROID**
Michal Pokorný
Android se v našich mobilech, tabletech, netboocích a ledničkách zabydlel, a asi se v nejbližší době neodstěhuje. Na nějakém jednoduchém zadání si ukážeme, jak se pro něj programuje. Vlastní notebook s nainstalovaným Android SDK se bude hodit.
Předpoklady: Aspoň pasivní znalost Javy.

Ladění, testování a kvalita kódu *("Proč se kamarádi mračí, když vidí moje zdrojáky?")* **QA**
Michal Pokorný
Programování není zdaleka jenom o programování. Nejdříve si povíme něco o základních nástrojích pro hledání a opravu už existujících chyb – debuggerech. Až budeme mít dost záludností, přejdeme k lekci psaní „hezkého kódu" a přepisování „hnusného kódu" na „hezký" v nějakém známém jazyce. Nakonec si povíme o správném testování a dalších nástrojích, které pomáhají vyvíjet kvalitní software.
Předpoklady: Aspoň jeden jazyk s OOP

Teorie množin *("Jablka a hrušky se dají nejen sčítat, ale třeba i násobit.")* **TEMNO**
Lukáš Folwarczný, Martin Mareš
Základoškolský přístup „množina je kupříkladu miska jablíček" nabízí spoustu otázek: Když jablíčka přesuneme do sáčku, bude to stále tatáž množina? A co když kousek jablíčka ukousneme? V rámci této přednášky se pokusíme o vybudování teorie množin od základů (rozuměj axiomů) a to v duchu Zermelo-Fraenkelovském. Pak uvidíme, jak na teorii množin vystavět zbytek matematiky.

Lineární algebra *("Vektorový prostor je místo, kde žijí vektory.")* **LA**
Jenda Hadrava
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?

Výběr vysoké školy *("Tady prý dobře vaří!")* **VS**
Karolína Burešová
Vybrat si vysokou školu může být náročný úkol. Čím se vlastně jednotlivé školy liší a čím z toho se obvykle moc nevytahují? Na co je dobré se při výběru ptát a co je spíš drobný detail? Souvisí nějak náročnost studia a vlídnost vyučujících? Povídání o rozdílech mezi školami od někoho, komu byla jedna škola málo.

Počítač bez myši *("Může počítač trpět fóbií z hlodavců?")* **KEYB**
Karolína Burešová
Ruce programátora patří na klávesnici, přesto se spousta věcí běžně dělá pomocí myši. Pojďme se podívat na programy, které s ovládáním klávesnicí počítají, a triky na ty ostatní. Spíše diskuse než přednáška, postřehy a zkušenosti všech zúčastněných jsou vítány.

Lingvištika *("Přísudek je v této větě podmět.")* **LING**
Martin Mareš
Převážně nevážné a mírně nepřed-vídatelné po-vídání o jazyku i jazyce. Základní jazykové rodiny a jejich podobnosti i odlišnosti. Co má společného čínština s angličtinou a co nikoliv. Jak se jazyky vyvíjejí a jak se navzájem ovlivňují. Kde jsme přišli k pravidlům a jaký je jejich smysl. Existují synonyma? Proč je jazyk nejednoznačný a proč je to dobře. Jak se na jazyk dívá matematik a jak se na matematiku dívají lingvisté. Jak vzniklo písmo? A jak otazník? Jak zapsat zachrochtání a jak třeba mlasknutí &c.

Literárně-filosofický seminář *("Filosofie a Matfyz, to přeci nemůže jít dohromady!")* **LFS**
Lukáš Folwarczný
Malé diskuzní posezení. Můžeme se pobavit o tom, jaká je úloha filosofie na Matfyzu. Vyměníme si tipy na zajímavé knížky, pomluvíme své oblíbené filosofy a pohádáme se nad velkými i malými otázkami: Spočívá tragédie soudobé společnosti v tom, že se nečte poezie? Dá se vůbec ve světě, kde lze zažít tolik zajímavých věcí, najít čas na beletrii? Kdo měl pravdu ve sporu o universália? Jsoucno, nebo bytí? Jaká je aplikace Platónova trojúhelníku v geometrii a jak souvisí Kant s Eulerem? A co na to Jan Tleskač?

Typografie *("What You See Is all What You've Got!?")* **TYPO**

Martin Mareš

Jak na počítači text nejen napsat, ale také vysázet tak, aby pěkně vypadal a aby (což je důležitější) se i příjemně četl. Jak se sází pohádka, jak báseň a jak vzorové řešení KSP plné komplikovaných vzorců. Jak jde dohromady staleté umění typografické a moderní technika. Přineste knihy i letáky, zkritizujeme sazeče, co se do nich vejde.

Orientace **ORI**

Martin Mareš, Jirka Setnička

Jak ze neztratit v terénu a jak se neztratit na moři. Vývoj umění navigace. K čemu je důležité slunce a hvězdy, ale proč mořeplavcům nestačí, alespoň dokud neobjevíme hodinky. Použití mapy, busoly a GPSky. Orientace bez pomůcek a použití Ariadniny nitě. Bleskový úvod do sférické astronomie a časomíry čili jak (ne)postavit sluneční a třeba i měsíční hodiny. Jak reprezentovat mapu v počítači a jak raději ne. Jak zapisovat polohu místa na Zemi (přestože Země má tvar podivně nakousnuté hrušky) a kolika způsoby to jde. Různé druhy map a jejich (z)kreslení. Jak se neztratit v kartografii. Praktické cvičení v terénu.

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.

Auto z pohledu technika *("Co mi to vrčí pod kapotou a proč bliká ta kontrolka?")* **CAR**

Jirka Setnička

Nahlédneme do tajů starších i novějších aut. Podle zájmu se můžeme pobavit o tom, jaký je rozdíl mezi benzínovým a naftovým motorem, či proč se auta staví zrovna tak, jak se staví. Na praktické ukázce probereme (a trochu rozebereme) auto a co nejvíce si ukážeme – z pohledu běžné údržby i jednoduchých oprav. Určeno pro každého, koho čeká autoškola, nebo ho jen baví mechanika.

Umělá inteligence biologicky *("Nasimulujte si zvířátko, které práci odvede za vás!")* **UIBI**

Jan Škoda

Naučíme se, jak počítač něco naučit bez programování. Nasimulujeme si v počítači malé modely mozku známé jako neuronové sítě a povíme si, jak takové učení souvisí s matematikou. Také budeme hledat řešení problémů pomocí evoluce. Dají se křížit nebo mutovat auta a vytvořit tak rychlejší?

Sociální epidemie *("Jak jsou nakažlivé myšlenky?")* **SOCEP**

Karel Tesař

Jak se šíří móda? Které reklamy jsou nejúspěšnější? Proč se mezi teenagery šíří kouření? Jací lidé jsou důležití pro vypuknutí sociální epidemie? To vše si řekneme na této přednášce.

Nevědomé myšlenky a první dojmy *("Jak moc řídíme to, jací jsme?")* **NEVMYS**

Karel Tesař

Výrazná většina činnosti našeho mozku jde mimo naše vědomí a my se podíváme pod pokličku, jak to funguje. Čím je ovlivněno naše vědomí? Jak je možné, že se můžeme hádat sami se sebou? Jsme to vždycky opravdu my? Jak důležité jsou první dojmy a kdy bychom jim měli věnovat pozornost?

Abecední seznam přednášek

LYK Stručný úvod do základů teorie vlkodlaků . . 1

Základní přednášky

OOP C++ a objektově orientované programování . 2
DYNP Dynamické programování 1
GA1 Grafy & algoritmy I 1
TEXT Hledání v textu 1
C Programování v jazyce C 2
PYTH Python 2
ZALG Základy algoritmizace, složitosti a datových struktur 1
ZAKL Základy programování 1
TEGE Základy teorie čísel a geometrie v algoritmech 1

Pokročilé přednášky

CPP C++ 3
BAR Čárové kódy 5
DS Datové struktury pro pokročilé 2
DIFF Derivace a integrály 5
DM1 Diskrétní matematika 5
GEOM Geometrie a počítače 2
AIGAME Herní algoritmy 2
HRAD Hradla 4
ITREE Intervalové stromy 2
SQL Jazyk SQL 3
ZIP Komprese dat 4
CRYPT Kryptologie 5
PST Matematika náhody 5
KOMP Od zdrojáku k programu 4
PARSE Parsing čili analýza textu 2
PERL Perl 3
GFX Počítačová grafika 3
CMPLING Počítačová lingvistika 4
HW Principy počítačů 4
CIS Programování v jazyce C# 3
JAVA Programování v jazyce Java 3
LOGP Prolog 3
RUBY Ruby 3
NET Sítě a Internet 4
GIT Systém pro správu verzí Git 4
TMP Teorie metrických prostorů 5
NONEX Teorie nemožného 5
TUR Testování uživatelského rozhraní 4
TEX TEX 4
HARD Těžké problémy 3
TOKY Toky v sítích 2
UNIX UNIX 4
WWW Webové stránky 3

Půlnoční přednášky

CAR Auto z pohledu technika 7
QA Ladění, testování a kvalita kódu 6
LA Lineární algebra 6
LING Lingvištika 6
LFS Literárně-filosofický seminář 6
NEVMYS Nevědomé myšlenky a první dojmy 7
ORI Orientace 7
KEYB Počítač bez myši 6
ANDROID Programování pro Android 6
SOCEP Sociální epidemie 7
TEMNO Teorie množin 6
TYPO Typografie 7
UIBI Umělá inteligence biologicky 7
SOUT Úvod do programátorských soutěží 6
VS Výběr vysoké školy 6
ZDRAV Základy první pomoci 7