Témata zápočtových programů

Verze 2.6 (Martin Mareš, 2011-02-11)

http://mj.ucw.cz/vyuka/zap/

Knihovny funkcí

Matice (5/5)
: Implementace běžných operací s maticemi (sčítání, násobení, inverze, determinant, LUP-dekompozice) použitím efektivních algoritmů. Doporučuji nahlédnout do knihy L. Kučery a J. Nešetřila Algebraické metody diskrétní matematiky.

Dlouhá čísla (5/5)
: Implementace běžných operací s dlouhými čísly (sčítání, odčítání, násobení, dělení, umocňování, vstup a výstup v libovolné číselné soustavě) efektivními algoritmy. Viz též Donald Knuth: Seminumerical algorithms (The Art of Computer Programming, Volume 2). *(6/6):* Reprezentace čísel pomocí reziduí (zvolí se n navzájem různých prvočísel a každé číslo se zapíše jako vektor zbytků po dělení těmito prvočísly; výhody: extrémně efektivní sčítání, odčítání a násobení – vše po složkách; nevýhody: obtížný převod do tohoto tvaru a zpět, jakož i dělení). Opět viz Knuth.

Stromy (5/5)
: Implementace operací s vyváženými vyhledávacími stromy (např. AVL, červeno-černými nebo 2-3): insert, delete, find, findmin, findmax, next.

Graph (5/7)
: Grafická knihovna pro FreePascal, schopnostmi srovnatelná se standardní unitou graph v Turbo Pascalu. Základní grafická primitiva, jako jsou úsečky, kružnice a oblouky, vyplňování oblastí a jednoduché vypisování textu. Rovněž by se hodila práce s myší.

Grafy (5/6)
: Knihovna pro práci s grafy orientovanými a neorientovanými (a třeba také s multigrafy nebo hypergrafy) – definuje si nějaké rozumné uložení grafu v paměti (třeba pole vrcholů a seznamy hran z vrcholů vycházejících) a nabízí funkce pro běžné grafové operace (např. hledání nejkratší cesty, minimální kostry, komponent souvislosti, transitivní redukce či uzávěru, topologické třídění grafu).

Databáze (6/6)
: Jednoduchá databázová knihovna organizující záznamy, které se skládají z klíče a hodnoty (vesměs libovolných délek). Záznamy jsou uložené v souboru na disku a knihovna nabízí efektivní (tj. v logaritmickém čase) přidávání a odebírání záznamů, procházení celé databáze a hlavně vyhledávání podle klíče. Na to se hodí například hashování nebo B-stromy (to jsou 2-3-stromy pro větší hodnoty 2 a 3).

Kombinatorika a jiné kratochvíle

Optimal sorting (5/6)
: Generátor třídícího algoritmu pro N prvků, který používá minimální možný počet porovnání. Viz Donald Knuth: Sorting and Searching (The Art of Computer Programming, Volume 3).

Hádankářský slovník (3/3)
: Program, který si načte slovník (co řádka, to slovo) a pak odpovídá (rozumně rychle) na dotazy chtivých hádankářů, které přesmyčky (matematik by řekl permutace) daného slova ve slovníku jsou.

Křížovky (6/6)
: Generátor křížovek – zadáte slovník a velikost křížovky, případně obrazec a nějaké požadavky na pevné rozmístění rozdělovacích znamének a program vygeneruje nějakou křížovku z daných slov v daném obrazci. Další nápady: požadavky na symetrii obrazce a "rozdělítek" (u klasických křížovek to je zvykem, stejně tak se v nich nesmí vyskytovat jedno slovo vícekrát, byť s různou legendou), ornamentální, slabikové, střídavé a jiné křížovky, výstup hotového obrazce v PostScriptu.

Sudoku (5/4)
: Vytvoří Sudoku zadané obtížnosti (míra obtížnosti je nutně heuristická, její stanovení je součástí úkolu). Možná rozšíření: hexadecimální sudoku, KenKen.

Skoky po šachovnici (3/3)
: Programu dáte definici šachové figurky (to jest nějak popíšete, jak se daná figurka může pohybovat), velikost šachovnice a souřadnice počátečního políčka a on vymyslí, zda a jak je možno touto figurkou "obskákat" celou šachovnici (to jest navštívit každé políčko právě jednou) a skončit přesně tam, kde jsme začali.

Polyomino (5/4)

Pentomino je tvořeno obrazci složenými z pěti jednotkových čtverečků, které sice nemusí být konvexní, ale musí být souvislé (tj. nemohou obsahovat díry). Napište program, který nalezne všechny obrazce skládající se z n čtverečků. Obrazce lišící se pouze otočením nebo zrcadlením se za různé nepovažují.

Skládačka (5/5)

Jsou zadány dílky složené z jednotkových čtverečků (třeba ty z předchozí úlohy) a máte zjistit, zda z nich je možno složit obdélník dané velikosti. Dílky je možno otáčet a překlápět.

IQ! (5/6)

Řešítko na klasické IQ-testové úložky typu "doplňte následující posloupnost". Mělo by znát běžné finty (sčítání a násobení předchozích členů, prvočísla apod.) a možná si též vést katalog již známých posloupností a odvozovat nové z nich.

Jede vláček motoráček... (6/7)

Hledání optimálního spojení v železniční síti nebo třeba v městské hromadné dopravě [tam není dobré spolehat se na jízdní řády – spíše uvažujte o intervalech či středních hodnotách čekání; nezapomeňte, že ve všech našich městech jezdí i pěškobus]. Existuje spousta možností, jak si hledátko vylepšit: například volitelná optimalizace na délku doby jízdy, doby čekání [na cesty v zimě], počet přestupů [pokud jste proflámovali noc] či cenu [pozor, u pásmového tarifu Českých drah se občas cenově vyplácí rozložit úsek cesty na více kratších].

Geometrie

Nejbližší dvojice (3/3)

Je dáno N bodů v rovině. Program vypíše souřadnice dvou bodů, které si jsou nejblíže (v Euklidovské metrice). Výpočet by měl být v lepším čase než kvadratickém s počtem bodů.

Mnohostěny (5/6)

Polopravidelný mnohostěn je konvexní těleso, jehož všechny stěny jsou tvořeny dvěma typy shodných pravidelných mnohoúhelníků a ve všech vrcholech se stýká stejný počet hran. (Každé pravidelné těleso je tedy současně polopravidelným.) Napište program, který dostane na vstupu parametry tělesa (typy a počty stěn a stupně vrcholů) a poté těleso zobrazí buďto v nějakém rozumném průmětu nebo jako rovinnou síť (dle vlastního výběru).

Mongeovský invertor (6/6)

Určitě všichni znáte Mongeovo promítání – to je takové to promítání trojrozměrných objektů na 3 kolmé průmětny (nárys, půdorys, bokorys), které používají strojaři a architekti. Opačná úloha je ale zajímavější: dostanete zadané všechny tři průměty nějakého tělesa a program má toto těleso zkonstruovat (často bude existovat více variant, tehdy stačí objevit jednu).

Fyzika

Problém mnoha těles (5/5)

Program simulující pohyb N těles navzájem na sebe působící gravitační silou. Grafický vstup, editor na polohy těles, proměnlivá rychlost simulace a velikost kroku.

Sky Globe (5/5)

Hvězdná mapa. V souboru jsou uloženy polohy hvězd na nebeské sféře (v rovníkových souřadnicích), program dostane zadané místo na Zemi, datum a čas a zobrazí oblohu tak, jak je v daný okamžik vidět. Další nápady: korekce na atmosférickou refrakci, paralaxu, precesi a nutaci, počítání východů a západů hvězd, kreslení hranic a tvarů souhvězdí atd.

Heliocentrický simulátor (5/5)

Simulace pohybu planet ve sluneční soustavě a zobrazení v ekliptikálních souřadnicích (to jest něco jako pohled shora). Výpočet poloh planet buďto podle Keplerových zákonů nebo přímo z Newtonových zákonů simulací problému N těles. Další nápady: proměnlivost dráhových elementů v čase (vzorce dodám), dráhy komet.

Geocentrický simulátor (8/7)

Simulace pohybu planet, ale s pohledem z daného místa na Zemi v daný okamžik. Též může počítat východy, průchody poledníkem a západy. Viz předchozí dvě úlohy.

Kmitání (4/5)

Simulace harmonických oscilátorů (tlumené kmity, resonance apod.). *(5/6):* Verze s hezkým grafickým výstupem.

Chemické rovnice (5/5)
: Vyčíslování chemických rovnic – uživatel zadá reakci a program dopočítá, kolikrát se který reaktant a produkt v rovnici má objevit, aniž by se ztrácela či naopak materializovala hmota.

Čočky (5/5)
: Simulace optických soustav – uživatel zadá systém čoček a zrcadel a program spočítá průchod paprsků tímto systémem (nebo třeba pohled tímto systémem na obrázek umístěný v nějkém bodě). *(7/7):* Částečné odrazy, polopropustná zrcadla, disperze světla, rozptyl, ...

Malý dráteník (6/6)
: Stavebnice elektrických obvodů – uživatel si ze základních součástek a vodičů vytvoří elektrický obvod a program simuluje jeho činnost. V jednoduché variantě třeba pouze digitální obvody (hradla, klopné obvody, zdroje signálů a indikátory), složitěji třeba obvody analogové s odpory, kondenzátory, cívkami a transistory.

Orloj (6/7)
: Simulátor orloje, třeba toho staroměstského.

Interpretery a kompilátory

Basic (6/4)
: Interpreter rozumné podmnožiny jazyka Basic (nebo nějakého jiného jednoduchého programovacího jazyka dle vlastního výběru).

LISP (6/5)
: Interpreter LISPu (čtení a vypisování LISPovských objektů, interpretace [tedy vlastně funkce eval], garbage collector).

CPU Simulator (5/4)
: Program pro simulaci nějakého jednoduchého procesoru (vhodnými příklady jsou např. ARM, Sparc či R4000, ale můžete si vymyslet i procesor vlastní, rád s tím poradím), to jest program, který načte program ve strojovém kódu příslušného procesoru a poté jej provádí a umožňuje si nechat vypisovat, co se zrovna děje (obsah registrů a paměti plus právě prováděné instrukce).

  K této úloze existuje spousta různých zajímavých vylepšení: implementace vstupu v assembleru, "vyspělejší" debugger (podporující podmíněné breakpointy apod.), simulování okolního hardwaru (kupříkladu řadič přerušení, časovače a nějaké porty) či memory manageru.

Parallel CPU Simulator (6/5)
: Simulace paralelního počítače na bázi předchozí úlohy, tentokráte již ovšem značně zjednodušující reálný svět. Dodefinujeme instrukce pro práci se semafory (init, lock, unlock; semafor je dán svou adresou v paměti), ve speciálním registru je pevně uloženo číslo procesoru, všechny procesory mají sdílenou paměť (nesmí jich ovšem zapisovat v daném okamžiku více na touž adresu či na ni jeden zapisovat a jiný z ní číst). Simulátor načte program v assembleru resp. strojovém kódu a simuluje jeho běh na všech procesorech současně včetně detekce kolizí přístupů k paměti a detekce vzniku deadlocku (procesy vzájemně čekající na uvolnění semaforů).

Compiler 1 (5/5)
: Kompilátor výrazů (syntaxí podobných Pascalu) do assembleru výše zmíněného simulátoru (možno i paralelizovat!).

Compiler 2 (7/6)
: Kompilace nějakého jednoduchého vyššího jazyka do "simulátorového assembleru". Možno implementovat některé základní optimalizační techniky (což může dát úlohu obtížnosti až 9).

Compiler 3 (8/6)
: Kompilace nějakého jednoduchého vyššího jazyka do assembleru paralelní verze "simulátorového assembleru" s rozumným využitím paralelismu. Opět možnost optimalizací.

The Last Optimizer (7/7)
: Odvěkým snem všech programátorů je napsat program, jenž k danému programu nalezne jiný, který dělá totéž, zato však co nejefektivněji. Dá se poměrně snadno dokázat, že to není algoritmicky řešitelný problém. Ovšem zkusme si úlohu zjednodušit tak, že budeme pracovat s naším "simulátorovým procesorem", vstupy omezíme na vybrané registry procesoru, zakážeme skoky dozadu (tedy vlastně cykly) a omezíme se na maximálně 20 instrukcí zdrojového programu a maximálně 10 cílového. Doporučená implementace: generování všech možných cílových programů od nejkratších k nejdelším a heuristické porovnávání funkčnosti s algoritmem původním (to znamená: generuji spoustu náhodných vstupů a porovnávám, že jsou výstupy tytéž). Viz GNU SuperOpt.

Manipulace s programy

Indent (6/5)
: Program pro formátování zdrojových textů v Pascalu či C. Automatické odsazování podle blokové struktury programu, formátování a správné umísťování komentářů. Viz UNIXový program téhož jména.

XRef (6/6)
: Program pro generování křížových referencí pro Pascal či C. Pro každý vstupní program (pokud možno včetně unit či modulů) vygeneruje kompletní tabulky odkazů na všechny konstanty, proměnné, procedury i funkce. Dá se též doplnit (7-8/8) o interaktivní interface umožňující rovnou z editoru vyhledávat odkazy na jméno právě se nacházející pod kursorem atd.

Pretty-printer (6/6)
: Program pro sazbu zdrojových textů programů v Pascalu či Céčku tak, aby se daly příjemně číst. Vítanou možností je konverze do TEXu, HTML nebo PostScriptu.

Vývojáci (6/4)
: Program překládající jednoduchou podmnožinu nějakého programovacího jazyka do vývojových diagramů.

GPerf 2 (6/7)
: Program na generování perfektních hashovacích funkcí. Na vstupu je tabulka párů klíčové slovo + hodnota, na výstupu je funkce v Pascalu či C realizující co nejrychlejší mapování klíčových slov na hodnoty, nejlépe metodou perfektního hashovaní. Viz dokumentace od programu GPerf, nicméně vyřešit rozumněji – počítat s tím, že některé klíče se mohou lišit pouze permutací písmen.

Interaktivní programy

Textový viewer (4/3)
: Jednoduché "prohlížítko" na textové soubory. Zadá se mu jméno souboru a ono nechá uživatele listovat si obsahem souboru. Nechť podporuje posun doleva, doprava, nahorů i dolů, jakož i po stránkách a hledání řetězce v souboru a nezalekne se netištitelných znaků ani milionu řádků.

HTML viewer (5/5)
: Prográmek na prohlížení dokumentů v jazyce HTML, včetně "inteligentního" formátování odstavců apod. Možno přidávat spoustu funkcí a dotáhnout to až na webový browser…

Textový editor (5/4)
: Jednoduchý textový editor na soubory, které se celé vejdou do paměti (ovšem bez omezení na počet řádků či délku řádku!). Běžné editační funkce: pohyb v textu, pohyb po slovech, delete slova či řádku, skok na řádek daného čísla, hledání a nahrazování textu, save, load, operace s bloky. (Dále třeba formátování odstavců, automatické odsazování, makra, …) *(7/7):* A co když se do paměti nevejdou?

Twilight Commander (6/4)
: Něco jako Norton Commander, resp. Midnight Commander – to jest něco, čemu se z příčin, které jsou dosud předmětem vyšetřování, v českých zemích říká souborový manažer. Konkrétněji: program poskytující komfortní rozhraní k operacím se soubory a adresáři.

Blbenka čili inteligentní diář (5/5)
: Připomínátko (jinak též reminder alias blbenka) je program, kterému píšete seznam událostí spolu s časy, kdy mají nastat (tedy takový plánovací kalendář) a ono vám je s volitelným předstihem připomíná (můžete si jej třeba pouštet každý den při zapnutí počítače). Spousta možností k rozšíření: periodické události (to jsou třeba narozeniny vašich bližních), zadávání pomocí parametrů (např. "každý pátek třináctého" nebo "každé úterý v 10:30"), triggery (přednastavíte si událost typu "vyndat oběd z trouby", která se objeví 5 minut po spuštění, přičemž ji pak v libovolném okamžiku můžete třeba stiskem nějaké předem přiřazené klávesy spustit; to je taková zobecněná minutka), deník (můžete připomínátku psát, kdy jste na čem pracovali, a ono vám pak spočítá, kolik času jste čím strávili).

Kdy my tři sejdeme se (6/7)
: … v dešti a hromobití? Tak se ptaly čarodějnice v Macbethovi a my jim zkusíme pomoci programem na domlouvání schůzek. Program si pro každého uživatele eviduje jeho časové možnosti (například to může být založeno na diáři z předchozí úlohy) a kdokoliv má možnost pozvat ostatní na dostaveníčko a prostřednictvím programu se s nimi dohodnout na termínu schůzky. Také by z toho mohla být pěkná webová aplikace.

Abacus (4/5)
: Simulace kuličkového počítadla nebo třeba logaritmického pravítka.

Notičky (6/7)
: Editor notových zápisů (noty, pomlky, ligatury, takty, slova písně etc.), třeba i se zvukovým výstupem. Nápady: Automatická harmonizace v zadané stupnici.

Grafika

Kreslítko (5/5)
: Program pro kreslení bitmapových obrázků. Základními operacemi jsou kreslení bodů, čar a kružnic, nastavování barev a tlouštěk čar a načítání a ukládání obrázků v nějakém standardním formátu (třeba PCX, GIF či nejlépe PNG). Další funkce: vyplňování oblastí, práce s bloky, filtry, volitelné zvětšení, editování obrázků větších než obrazovka, psaní textu, transparentní a polotransparentní barvy, zvětšóvání, zmenšování a otáčení obrázků.

Vektorové kreslítko (6/7)
: Opět kreslící program, tentokráte na obrázky vektorové, tj. skládající se z úseček, křivek (například Bézierových) a šrafovaných ploch a nezávislé na rozlišení výstupního zařízení (tak se vytváří například většina obrázků v knížkách, pokud to zrovna nejsou reprodukce fotek). Definice vlastních grafických prvků pomocí prvků elementárních a dříve nadefinovaných; práce se symetriemi, snapping (když uživatel ukáže "někam do neznáma", vybrat si nejbližší významný bod (vrchol, průsečík, řídící bod křivky, bod mřížky apod., tak, jak uživatel nastaví), export do PostScriptu či MetaFontu. Vydatnou studnicí nápadů je editor IPE (předvedu).

Editor na grafy (5/5)
: Program pro interaktivní editaci orientovaných i neorientovaných grafů. Ukládání do souboru v rozumném formátu (seznam vrcholů a hran či něco podobného) a načítání zpět. Ovladatelné klávesnicí i myší.

Advanced graph editor (8/7)
: Opravdu dobrý grafový editor schopný pro daný graf nalézt rozumné nakreslení (minimalizující počet křížení) a sám volit routing hran. Rozumný export grafu do PostScriptu či MetaFontu.

Grafy funkcí (5/6)
: Uživatel zadá funkci, program nakreslí její graf, přičemž inteligentně zvolí měřítko a popíše osy. Manuální ovládání – zoom vybrané části grafu apod. *(6/6):* Trojrozměrné grafy (funkcí dvou proměnných nebo třeba komplexních funkcí).

Trojrozměrný viewer (5/4)
: Program načte ze souboru popis mnohostěnu (to jest souřadnice vrcholů, které vrcholy tvoří hrany a které hrany tvoří stěnu; konvexnost se nepředpokládá) a zobrazí jej v nějakém "rozumném" průmětu (volné rovnoběžné promítání či promítání na kolmou průmětnu – může jich být i více). Uživatel si může tělesem otáčet a prohlížet si ho z různých stran. *(6/5):* Stínování stěn nebo generování stereogramů (to jsou takové ty "šilhací" trojrozměrné obrázky).

Interaktivní geometrie (5/7)
: Program, který na základě popisu v nějakém jednoduchém jazyce (vzpomínáte si na zápisy konstrukce ze základní školy?) provádí rovinné geometrické konstrukce ze zadaných bodů a zobrazuje je. Mohl by také dovolit uživateli, aby myší pohyboval zadanými body a sledoval při tom, jak se výsledek mění.

Interaktivní stereometrie (7/7)
: Trojrozměrná verze předchozí úlohy.

Interaktivní hypergeometrie (8/8)
: N-rozměrná verze předchozích dvou úloh. Zkuste vymyslet vhodné promítání, ve kterém by se daly rozumně pozorovat alespoň čtyřrozměrné, lépe i vícerozěrné objekty.

Alternativní geometrie (7/7)
: Ne každá geometrie je euklidovská – existují i jiné, třeba sférická (ta se odehrává na sféře, tj. povrchu koule) a hyperbolická (tu potkáte např. v horských sedlech). Jsou v mnohém podobné, ale třeba úhly v trojúhelníku mohou v takové geometrii dát dohromady klidně více (ve sférické) nebo méně (v hyperbolické) než 180 stupňů. To je na první pohled velice neintuitivní, ale kupodivu mnoho matematických i fyzikálních teorií (za všechny jmenujme alespoň obecnou teorii relativity) se snadno popisuje právě v takovýchhle "divných" geometriích, a tak stojí za to si (třeba právě na interaktivním počítačovém simulátoru) chování sférického nebo hyperbolického světa osahat.

Interpolace splinami (4/4)
: Mějme N bodů v rovině. Program je zobrazí a proloží jimi spline-funkci (to je funkce, která je mezi každými dvěma sousedními body nějakým polynomem třetího stupně – typicky pokaždé jiným – a je spojitá až do druhé derivace; druhá derivace v okrajových bodech je 0; toto ji určuje jednoznačně). Ruční pohybování body za současného překreslování proložené křivky není rozhodně na škodu. *(6/5):* Prostorové spliny (splinové pláty).

Interpolace polynomem (4/4)
: Jako minulá úloha, až na to, že prokládáme jedním polynomem stupně k, například metodou nejmenších čtverců (na požádání vysvětlím).

Bézierovy křivky (4/4)

Jako minulá úloha, až na to, že neprokládáme, nýbrž zobrazujeme Bézierovu křivku určenou danými body. (Popis najdete např. ve Wikipedii.)

Vzorečky (6/5)

Pohledná sazba vzorečků: uživatel zadá nějaký vzorec (s indexy, exponenty, zlomky, závorkami, funkcemi, sumami a třeba i limitami a integrály) a dostane se mu graficky hezky vyvedené podoby tohoto vzorce. Jako inspirace může dobře posloužit program TEX.

Formuláře (6/7)

Program pro vyplňování formulářů a podobných lejster: dostane oscanovaný formulář, najde si v něm čáry a podle nich kolonky, nechá uživatele kolonky upravit a popsat a výsledek si uloží jako šablonu, do které pak půjde doplňovat data a tisknout tak, aby vyšla přesně do kolonek na papíře.

Matematika

Čtyři čtyřky (3/4)

Napište program, který pro všechna přirozená čisla od 0 do N zjistí, jak je zapsat za pomoci čtyř čtyřek (respektive jiného počtu jiných číslic dle zadání uživatele) a běžných aritmetických operací. Hint: zkuste použít dynamické programování.

Faktorizátor (6/5)

Rozklad velkých čísel na součin prvočísel chytrými algoritmy. Viz Donald Knuth: Seminumerical Algorithms (The Art of Computer Programming, Volume 2).

Velká prvočísla (5/5)

Generování velkých prvočísel (řádově stovky číslic) pravděpodobnostní metodou (viz L. Kučera: Kombinatorické algoritmy). Možno též využít malé Fermatovy věty (je-li p prvočíslo a x nesoudělné s p, je $x^{p-1} \bmod p = 1$), generují se náhodná x a testuje se, zda pro p toto tvrzení platí.

Náhodná čísla (4/5)

Program rozhodující o dané posloupnosti, zda má náhodný charakter. Viz metody popsané v Seminumerical Algorithms.

Kalkulačka (4/3)

Simulace jednoduché kalkulačky s bežnými operacemi včetně základních funkcí (sin, cos, ln, exp, sqrt atd.), závorek a předností operátorů. Pozor na správné ošetření chyb (dělení nulou apod.).

Výrazová kalkulačka (4/4)

Uživatel píše výrazy a program je vyhodnocuje. Opět základní funkce, závorky a priority operátorů.

Programovatelná kalkulačka (5/5)

Výrazová kalkulačka, v níž si uživatel může definovat vlastní funkce a operátory, případně i typy (to znamená například komplexní čísla či quaterniony [4-složková podoba komplex. čísel, existují 3 imaginární jednotky, $i^2 = j^2 = k^2 = -1$, $i \cdot j = k$, násobení není komutativní], nebo dokonce zlomky a intervaly). Zajímavým zdrojem inspirace může být program Calc pro unixovský editor Emacs.

Úpravy výrazů (7/6)

Program na operace s výrazy (zjednodušování, roznásobování a derivování). Nejlépe interaktivně; záhodno spojit s výrazovou kalkulačkou.

Lineární rovnice (3/3)

Řešení soustav lineárních rovnic (zadaných textově, ne pouze jako matice).

Nelineární rovnice (6/6)

Řešení nelineárních rovnic a jejich soustav, například nějakou šikovnou numerickou metodou.

Kostkové vzorce (5/6)

V mnoha hrách na hrdiny se objevují vzorce typu `XdY`. To znamená "Vezmi kostku o `Y` stěnách, `X`-krát s ní hoď a výsledek sečti." Jednoduchý vzorec stylu `2d4`, `3d6+1`, `2d10+1d20-5` se ještě dá pochopit a odhadnout, ale jak se chová vzorec `(2d3-2)d(3d2)`, to už se určuje opravdu těžko. Napište program, který na vstupu přečte takovýto vzorec (s operátory `+-,*,d` v tomto pořadí priorit vzestupně, pro machry i dělení) a vypíše pro každý možný výsledek pravděpodobnost, s jakou nastane.

Jazyky a texty

Rozpoznání jazyka (5/7)

Rozhodnutí, v jakém jazyku (řekněme z pěti známých) je zadaný text. Statistické metody (pravděpodobnosti výskytu dvojic znaků nebo něco podobného).

Markovovské texty (4/5)
Generování náhodného textu napodobujícího text zadaný (řekněme se stejnými pravděpodobnostmi výskytu znaku v závislosti na předchozích k znacích). Případně totéž pro slova. [Lze též řešit způsobem připomínajícím Huffmanovu kompresní metodu, na požádání předvedu.]

Spelling Checker (5/5)
Kontrola textu za pomoci slovníku známých slov. Interaktivní mód umožňující online doplňování do slovníku včetně uvedení skloňování slova (jako vzor je možno uvést libovolné již známé slovo). Důraz se klade na rychlost. Pokud objeví překlep, zkusí najít nejbližší známá slova.

Rychlé vyhledávání (4/4)
Na vstupu je seznam slov (až stovky) a velice dlouhý text. Program nalezne první výskyt některého ze slov v textu. Předpokládejte, že ve vstupním textu se nemůžete vracet (například proto, že to je výstup nějakého jiného programu).

Rychlé nahrazování (5/5)
Na vstupu je seznam až stovek dvojic slov a dlouhý text. Program v textu nahradí všechna první slova dvojic druhými z téže dvojice. Opět se nelze ve vstupním souboru vracet.

Špión (5/4)
Luštění tajných abeced. Je zadán text (poměrně dlouhý a ve známém jazyce) zašifrovaný neznámým šifrovacím algoritmem (z nějaké dostatečně jednoduché skupiny šifer, řekněme těch, které nahrazují každý znak pevně zvoleným znakem nezávisle na tom, kde se v textu vyskytl – tj. změna abecedy) a náš program na základě známých statistických vlastností jazyka (např. pravděpodobnosti výskytu jednotlivých písmen a dvojpísmenných kombinací) nalezne nejpravděpodobnější znění původního textu.

Sazba HTML (6/6)
Úhledné vysázení webové stránky, například převodem do TEXu. Alespoň základní tagy, i když tabulky by také byly pěkné.

Zlutoucky kun (6/7)
Program se natrénuje na velkém množství textů a pak bude do textů bez diakritiky doplňovat háčky a čárky.

Slabiky (5/7)
Rozklad českého textu na slabiky (to se hodí třeba při sázení zpěvníku).

Hry (proti počítači)

Pišqorky (4/1)
Klasická hra *piškvorky*. Statické ohodnocování možných tahů včetně možné reakce soupeře.

Pišqorky 2 (5/4)
Chytřejší implementace hry *piškvorky* – prohledávání stavového prostoru hry na zadaný počet tahů dopředu, minimaxový algoritmus založený na statickém hodnocení pozic překračujících hloubku prohledávání.

Pišqorky 3 (7/5)
Předchozí úloha doplněná cache-pamětí na již zpracované situace. Teoreticky prozkoumat a případně i implementovat možnost normalizace pozic před hledáním v cache.

Tic-Tac-Toe (2/1)
Piškvorky na hracím plánu 3 krát 3 políčka, vítězí hráč, který dosáhne třech kamenů své barvy v řádku, sloupci či úhlopříčce. Vyžaduje se použití optimální strategie.

Reversi (5/4)
Hra *Reversi* má následující pravidla: Hraje se na klasické šachovnici, na počátku jsou uprostřed hracího plánu položeny dva černé (na D4 a E5) a dva bílé (na E4 a D5) kameny. Hráči se střídají v pokládání kamenů své barvy, při položení kamene na dané políčko se všechny nově vzniknuvší kombinace (v řádku, sloupci či úhlopříčce) dvou kamenů právě táhnoucího hráče, mezi nimiž jsou (bez prázdných políček) kameny druhého hráče, přebarví na barvu táhnoucího hráče. Položení kamene, po kterém by nenásledovalo žádné přebarvení, není dovoleno. Hra končí, nemůže-li hráč, který je zrovna na tahu, nikam položit další kámen. Vyhrává ten, komu patřilo na konci hry více kamenů.

Reversi 2 (7/5)
Chytrý algoritmus na *Reversi* – prohledávání na více tahů do hloubky, cache paměť zabraňující zbytečnému přepočítávání stejné situace vícekrát, normalizace situací zabraňující opakovaným výpočtům ekvivalentních situací (všimněte si, že situace lišící se jen otočením, zrcadlením a několika dalšími transformacemi šachovnice jsou ekvivalentní).

Exploding Atoms (5/4)

Hra se hraje na normální šachovnici. Na každém políčku je jedno atomové jádro, které může patřit jednomu z hráčů a mohou kolem něj obíhat elektrony. Na počátku hry nepatří žádné jádro nikomu a nikde nejsou žádné elektrony. Hráči se střídají, v každém tahu jeden z nich přidá elektron k některému ze svých atomů, případně k atomu dosud neobsazenému (který si tím přivlastní). Pokud tak počet elektronů dosáhne kritického množství (to je rovno počtu sousedů daného atomu, tedy 2 pro rohové atomy až 4 pro vnitřní), atom exploduje a jeho elektrony se rozletí do všech směrů k sousedním atomům, které tak rovněž připadnou táhnuvšímu hráči a případně také explodují atd. Hra končí, přijde-li jeden z hráčů o všechny své atomy (tím prohraje), případně dojde-li k zacyklení explozí (remíza, může vůbec nastat?). [Podobnost s fyzikálními zákony, pokud nějaká existuje, je čistě náhodná.]

Exploding Atoms 2 (7/5)

Lepší strategie na předchozí hru prohledávající na více tahů do hloubky a používající cache a transformace.

Logik (5/4)

Známá hra *logik.* Hra je parametrizována dvěma čísly: počet pozic N (typicky 5) a počet barev M (typicky 8). První hráč na počátku zvolí nějakou N-prvkovou kombinaci daných barev (mohou se i opakovat), druhý se ji snaží uhodnout. V každém tahu druhý hráč navrhne nějakou kombinaci barev a první mu na ni odpoví dvěma čísly: počet správně uhodnutých pozic a počet barev, které sice v utajované kombinaci jsou, ale na jiném místě (tedy například je-li tajná kombinace 12321 a druhý hráč hádá 14212, je odpověď 1/3). Od programu se očekává, že pro typické zadání bude schopen do deseti tahů a jedné minuty problém vyřešit.

Mefisto (5/4)

Táž hra, jako v předchozí úloze, tentokráte ovšem z pozice druhého hráče, který podle využívá toho, že jej nikdo nenutil, aby na začátku hry přiznal, co přesně zadal za kombinaci, a tak si nějak udržuje přípustné kombinace (to jest takové, které nejsou vyloučeny tím, jak v průběhu hry odpovídal) a vždy odpovídá nejškodoliběji, jak jen může (to jest snaží se co nejvíce hru protahovat a soupeře vhánět do úzkých).

Hangman (3/3)

Pravidla: program hádá slovo (za pomoci slovníku), jehož délka je předem známa. V každém tahu navrhne jedno písmeno a protihráč mu odpoví, kde všude se v hádaném slově vyskytuje. Čím méněkrát se splete, tím lépe.

Life (3/3)

Simulace života buněk. Je dán čtvercový životní prostor a v něm jsou buňky, které se vyvíjí v závislosti na svém okolí (např. buňka přežije do další generace, pokud má alespoň dva sousedy a naopak na prázdném políčku může vzniknout buňka, má-li více jak dva živé sousedy). Na vstupu je počáteční rozložení buněk a nějaká pravidla vývoje (buďto mezní počty pro přežití a množení nebo dokonce něco složitějšího v závislosti na rozsáhlejším okolí) a program pak simuluje generaci po generaci vývoj v celém prostoru. Další náměty: definování maximálního stáří buňky či třeba zkusit množící se buňky použít na řešení nějaké praktické úlohy (sčítání dvou čísel je krásný příklad); je zde spousta prostoru k optimalizacím pomocí bitových operací.

Mysli si zvíře (3/3)

Klasická hra z oblasti "umělé inteligence". Program zná nějaká zvířata a otázky na ano/ne ("Je to červenomodře pruhované?", "Živí se to kamením?", "Žije jen v okolí jaderných elektráren?"), pro každé zvíře též správné odpovědi na všechny otázky. Protihráč si myslí nějaké zvíře a počítač se jej snaží co nejmenším počtem otázek uhodnout.

Lloydova Patnáctka (4/4)

Další ze série poněkud antikvitních her. V krabičce velikosti 4 krát 4 se vyskytuje 15 jednotkových čtverečků (očíslovaných 1 až 15) a jedna unikátní jednotková díra. Cílem je posouváním (v daném kroku lze vždy na místo díry posunout libovolný sousední čtvereček, čímž vznikne další díra [nebo se ta stávající posune?]) čtverečky srovnat tak, aby čteno po řádcích byly očíslovány vzestupně.

Lloydova 255ka? (6/4)

Různé další variace na předchozí příkládek – vícerozměrná patnáctka, třeba i hyperkvádrová, patnáctka na anuloidu (to jest konce krabičky jsou slepeny k sobě), patnáctka na Möbiově proužku atd.

Šachové úlohy (5/6)

Řešení šachových úloh typu "je dána tato pozice, bílý táhne a dá třetím tahem mat".

Vláčky (6/7)

Simulátor železnice. Hráč si vytvoří kolejiště (tratě, výhybky, semafory, zastávky) a pak si po něm může pouštět vláčky (některé mohou třeba jezdit automaticky podle přednastavených jízdních řádů). Další nápady: simulace zabezpečovacího zařízení s automatickými semafory.

Code Wars (7/8)

Souboj robotů. Každý z hráčů si naprogramuje v nějakém jednoduchém jazyce svého robota a poté jsou

roboti vypuštěni do předem připraveného prostředí Každý z nich se začne chovat podle svého programu a jeho cílem je zničit všechny soupeřovy roboty – kdo přežije, vyhrává.

Člověče, nezlob se (4/4)
Počítačová verze známé deskové hry.

Pexeso (4/4)
Opět klasická desková hra: v rovině jsou rozmístěny kartičky obrázkem dolů, každý obrázek se vyskytuje právě dvakrát. Hráč, který je právě na tahu, otočí dvě kartičky. Pokud je na nich tentýž obrázek, obě získává, odebírá ze hry a otáčí další dvojici; pokud obrázky různé, otáčí je zpět a pokračuje další z hráčů. Hra končí, když jsou uhodnuty všechny páry obrázků; vyhrává ten hráč, který jich nasbíral nejvíce. Hra proti počítači by měla mít nastavitelnou obtížnost, která bude určovat pravděpodobnost, že počítač zapomene už jednou zpozorovaný obrázek.

Matfyzácké Pexeso (4/5)
Pexeso, v němž když hráč otočí dvě různé kartičky, vrací je zpět v opačném pořadí.

Deflektor (5/6)
Hra se hraje v bludišti, ve kterém je rozmístěn laser, terč a několik zrcadel. Cílem hráče je před uplynutím časového limitu otočit zrcadly tak, aby se paprsek z laseru trefil do terče; stěny místnosti mohou světlo buďto pohlcovat nebo naopak odrážet. Varianta pro dva hráče: Každý hráč má svůj terč, který má zasáhnout, hráči si otáčením zrcadel paprsek navzájem "kradou".

Červi (5/6)
Hra pro n hráčů, z nichž každý ovládá jednoho červa. Všichni červi lezou bludištěm, ve kterém jsou rozmístěny kousky jídla a jedu. Na počátku má každý červ délku jedno políčko, když objeví jídlo (tj. dolezou na políčko, na kterém nějaké je), prodlouží se o další políčko. Když červ sežere jed, narazí do zdi nebo do libovolného červa (včetně sebe sama), vypadává ze hry. Ovládání: Pro každého hráče nezávislé klávesy pro zatočení doleva a doprava. *(6/7):* Síťová verze.

Férové miny (5/5)
Implementace klasického minesweeperu se dvěma vlastnostmi navíc: Pokud uživatel, soudě dle stavu hrací plochy, nemohl mít ani u jednoho políčka jistotu, že na něm mina není (např. vždy na začátku hry), mina po odkrytí libovolného políčka na tomto políčku není. Naopak pokud uživatel mohl mít jistotu, že na některém políčku mina není, a přesto odkryl políčko, u nějž tuto jistotu mít nemohl, mina na něm je.

Puzzle (5/6)
Rozřeže obrázek na různě tvarované dílky a nechá si ho složit.

Tantrix (6/6)
Strategie na hru Tantrix (pravidla viz Wikipedia).

Soubory

Duplikáty (4/5)
Program prohledávající disk a zkoumající, jestli nějaké dva soubory nemají tentýž obsah (a to nějakou lepší metodou než je porovnávání každého souboru s každým).

Zápočťáky (6/7)
Program prohledávající disk a zobrazující podobné soubory (rozumné kriterium podobnosti souborů stanovte sami – dva textové soubory lišící se přidáním či změnou několika málo řádků zaručeně podobné jsou). Můžete předpokládat, že soubory jsou poměrně velké (typicky desítky kilobytů) a změny poměrně řídké.

Diff (6/5)
Nalezení rozdílů mezi dvěma textovými soubory. Výstup by měl obsahovat informace typu "zde vloženo těchto pět řádků, tamhle vymazáno těchto šest, onde změněn jeden".

Komprese (5/5)
Implementace nějakého pěkného (ať již klasického či vlastní provenience) kompresního (a k tomu též dekompresního) algoritmu.

Hledej, Šmudlo! (6/7)
Program, který vyřeší odvěký problém zmatku na disku. Bude umět hledat soubory podle zadaných kriterií – součástí jména, data poslední změny, textu uloženém v souboru atd. A aby to zvládl rychle, sám si soubory čas od času projde a z objevených dat si vytvoří nějaký jednoduchý index. Bylo by pěkné, kdyby uměl volat externí programy pro převádění různých formátů souborů (třeba PDF) na text.

Různé aneb co se jinam nevešlo

NetMapper (6/8)

Program na mapování sítí – prostřednictvím různých protokolů si "očuchá" topologii sítě a v přehledné formě ji vypíše, respektive nakreslí. (Může zahrnovat spoustu zajímavých věcí včetně odhadování propustnosti linek na základě závistlosti zpoždění na velikosti packetů – viz program Van Jacobsonův program pathchar).

JunkBuster (6/8)

Heuristický analyzátor e-mailů snažící se uhodnout (s co největší úspěšností, pochopitelně), které maily jsou nesmyslné reklamy a které nikoliv. Nejlépe adaptivní algoritmus schopný zkonstruovat si rozpoznávací pravidla sám, je-li mu poskytnut velký archiv s příklady reklamních a smysluplných mailů. Nabízí se celá řada triků z nejrůznějších oblastí, počínaje jednoduchým vyhledáváním podezřelých slov a frází v hlavičkách i těle mailu, konče třeba učením neuronových sítí.

Mraveniště (4/4)

Simulace jednoduchého světa – mraveniště je vlastně bludiště s místnostmi a chodbičkami, v něm jsou rozmístěny předměty (hlavně jídlo a stavební materiál) a pohybují se jím mravenci. Každý mravenec se řídí svým vlastním jednoduchým programem: to může být buďto nějaký klasický program nebo třeba pravděpodobnostní konečný automat (v každém okamžiku je mravenec v jednom z konečné množiny vnitřních stavů a jeho chování je popsáno tabulkou, která každé kombinaci stav + okolí [zda vidí zeď či nějaký předmět] s určitou pravděpodobností přiřadí nový stav a akci [krok, sežrání jídla, ...]). Možno experimentovat s chováním různých programů, se soutěžením mravenců navzájem (kdo přežije? Pokud ještě nadefinujeme rozmnožování mravenců, máme z toho už vlastně genetické algoritmy) atd.

Akordy (5/7)

Generátor kytarových akordů – pro akord zadaný jménem vymyslí, ze kterých tónů se skládá a nalezne všechny rozumné možnosti, jak jej na kytaru zahrát.

Papoušek (4/7)

Pouťový papoušek tahající z krabičky lístečky s věštbami, lépe řečeno jeho formální model. K dispozici je velké množství textů tvořících jednotlivé lístečky (třeba unixová databáze fortune cookies) a na vstupu je zadána nějaká věta. Program vybere lístek, který je zadané větě podle nějakého kriteria co nejpodobnější (například obsahuje nejdelší úsek textu společný se zadaným) a vypíše ho. Pro pobavení publika možno iterovat, tj. nechat na tento lístek nalézt další odpověď etc. ad libitum. Blázniviny: použití pro generování signatur e-mailů nebo třeba připojení do nějakého síťového chatu.