

Odkud přišla Algovize?

Povídání pro studenty

Rozšiřujícího semináře ADS na MFF UK

prof. RNDr. Luděk Kučera, DrSc.

algovision.org

a

MFF UK a FIT ČVUT

8. března 2021

I když jsem absolvoval MFF UK v oboru matematická analýza (informatika tenkrát jako samostatný obor neexistovala), začal jsem se pod vedením svých učitelů, hlavně Zdeňka Hedrlína, ale také Věry Trnkové a Aleše Pultra zajímat o teorii kategorií a o teorii grafů. Od kategorií jsem záhy utekl, ale zájem o grafy se zaměřil na grafové algoritmy a u algoritmů diskrétní matematiky jsem zůstal až do nynější doby.

Postupně jsem začal jako mladý asistent algoritmy diskrétní matematiky také vyučovat. Jelikož v té době byla cizojazyčná literatura těžko dostupná a česká prakticky neexistovala, napsal jsem knihu “Kombinatorické algoritmy” (Státní nakladatelství technické literatury, Praha 1983), která byla docela úspěšná, dostala cenu SNTL za rok 1983 a vyšla ve dvou vydáních česky a jednom anglicky (Hilger, 1990). V červnu 2002 studenti odevzdali rozpůjčené výtisky do knihovny v Karlíně, kde se pak v srpnu utopily při povodni.

V roce 1989 a letech následujících se v Československu hodně změnilo. Od velkých a největších věcí po drobnosti, jako byl seznam informatických přednášek na matfyzu. První roky devadesátých let byly dost živelné - myslím, že ku prospěchu věci. Informatika se celosvětově velmi rychle proměňovala, doba před 5-10 roky byla skoro pravěkem, a hodně nových přednášek na velmi aktuální témata se dostalo ke studentům prakticky ihned, bez zdlouhavého schvalování různými komisemi a kolegií.

Kolegové Michal Chytil a Honza Pavelka v té době napsali pro novou přednášku skriptu “Algoritmy”. Když pak Michal odešel do vládní komise a Honza do průmyslu, učili jsme jednoho roku tu přednášku ve dvou paralelkách s Alenou Koubkovou (která ten text s Honzou později upravila na Úvod do teoretické informatiky, Matfyzpress, Praha 1998). Mně se zdálo, že v přednášce je hodně o modelech výpočtu (to byl Michalův obor a Honza byl původně logik, než svůj odborný zájem otočil velmi radikálně k praktickým otázkám). A tak jsem ve své paralelce modelů výpočtu hodně ubral a přidal hodně algoritmů, veden snahou, aby byly z různých oborů a aby to byly metody, které jsou skutečně používané.

Tehdy takové úpravy šly bez velkého schvalování provést, ale ta změna byla už opravdu dosti závažná, vlastně dva různé předměty pod jedním označením. Kolegium inženýrského proděkana se začalo zabývat otázkou, kterou z těch dvou variant ponechat - a myslím, že rozhodlo docela dobře: přednášku rozšířilo na dva semestry, a dá se s jistými výhradami říci, že to dopadlo tak, že Chytil-Pavelka-Koubková dostali jméno ADS I a z mé varianty vzniklo ADS II.

Již v 80. letech se začaly objevovat grafické systémy, které z počítačů udělaly nejen chrlič rozmazaných písmen velké abecedy na obrazovku nebo v příšerné kvalitě prostřednictvím řádkových tiskáren na papír, ale i stroje, které uměly kreslit na obrazovku dynamické obrázky. A přišli lidé, kteří se pokoušeli této schopnosti využít k tomu, aby znázornili časový průběh výpočtu algoritmu. Už v roce 1983 vzniklo video R. M. Baecker, *Sorting out sorting*, prezentované na konferenci ACM SIGGRAPH, v roce 1984 vyšel článek M. H. Brown a R. Sedgewick, “A system for algorithm animation”, *Computer Graphics* 18(3), 177-186.

Opravdový boom animace algoritmů ale přišel v druhé polovině 90. let, když se objevila Java (a některé podobné systémy) a grafická práce s myší, okny, ikonami a kreslením na obrazovku přestala být doménou Macintoshů, neboť i Microsoft se od MS-DOSu s příkazovými řádky dostal k prvním Windows.

Počátkem nového tisíciletí už byla “algorithm animation” bouřlivě se rozvíjející obor se svými konferencemi a časopisy. Každý, kdo se v tom pohyboval musel mít svoji animaci binárního vyhledávacího stromu a/nebo Dijkstrova algoritmu, ba někteří se dostali i dále. Já jsem do rozjíždějícího se vlaku naskočil někdy v roce 1977 a pro přednášku ADS I-II (viz výše) jsem začal vytvářet applety v Javě, které v sobě měly více dynamiky než kroková či skoková animace klasických slajdů.

Animace algoritmů ale někdy po roce 2005 začala viditelně usychat (až nakonec uschla skoro úplně). V roce 2007 třeba vyšel o animacích článek R. Ben-Bassat Levy a M. Ben-Ari, “We work so hard and they don’t use it: acceptance of software tools by teachers”, 12th SIGCSE Conference. (Pracujeme tak tvrdě a oni to nepoužívají). “We” znamenalo my, kteří tvoříme animace, “they” byli učitelé algoritmů, kteří o ty animace neměli moc zájem.

Já (si) to vysvětluji na následujícím příkladu: v únoru 2005 jsem se dostal do Pittsburghu zrovna v den, kdy se hrál Super Bowl - finále ligy amerického fotbalu, asi největší sportovní událost v USA. Pittsburgh Steelers hráli proti Seattle Seahawks. Celé město bylo ověšeno nápisy “Steelers Go” a bylo společenskou povinností, že jsem se v hotelu ve svém pokoji na utkání také díval. Na rozdíl od baseballu je americký fotbal zajímavou podívanou i pro cizince, urostlí a rychlí mladí muži s vycpanými rameny běhají po hřišti s šišatým míčem, podrážejí si nohy a padají a vše je velice dynamické. Ale že jsem vůbec neznal pravidla, tak jsem nebyl nakonec ani schopen poznat, kdo vyhrál a jak moc. Teprve když jsem se zašel zeptat do lobby, z jásotu jsem poznal, že Steelers vyhráli (21-10).

A podobné je to s animacemi algoritmů: na obrazovce se míhají kolečka a obdélníčky, rudnou, zelenají, zvětšují se a mizí, ale jen ten, který dopředu algoritmus zná, je schopen poznat, co se tam děje, ostatní tam vidí jen chaotický rej obrazců. A uznejte, to není moc šikovné, když tou animací chcete někomu vysvětlit, jak (a dokonce i proč) ten algoritmus funguje. Není pak divu, že “they don’t use it”.

Takže se postupně ukazovalo, že užitek pro výuku algoritmů animace nemají ani zdaleka takový, jak se původně nadšeně doufalo. Problém byl i s Javou - je to skvělý jazyk s obrovskými možnostmi. I s obrovskými možnostmi zneužití pro tvorbu nejrůznějšího malware, takže nakonec různá bezpečnostní opatření způsobila, že applety v Javě už po internetu pořádně nešly spustit (pokud autor nepodstoupil komplikovanou a nákladnou certifikaci). Navíc i v rajských dobách na začátku bylo nutno si Javu předem nainstalovat, což sice nebylo složité, ale i tak omezovalo okruh potenciálních uživatelů.

A navíc v roce 2006 vážně onemocněla moje manželka, po několika letech ji nemoc zdolala, pak jsem potkal svoji nynější manželku - v kvůli tomu všemu ve vývoji Algovize zeje desetiletá díra.

Ale asi před čtyřmi roky Algovize chytila druhý dech. Naše počítače (včetně těch, které nosíme v ruce a říkáme jim chytrofony) se staly tak neuvěřitelně rychlými, že celkem uspokojivě zvládají i dosti pokročilou grafiku, třeba i tu, naprogramovanou v Javascriptu.

A Javascript je druhé kouzlo - ne že by to byl jazyk, ve kterém by se snadno programovalo a že by běžel hezky svižně, to vůbec ne. Vlastně se v něm dá programovat *velmi snadno*, dovolí Vám skoro vše, na co si vzpomenete; to se Vám pak ale vymstí až uděláte nějakou chybu. Kompilátor hodně z nich pustí a pokud nezachováte tvrdou disciplinu v programování, strávíte většinu času jejich hledáním a proklínáním nápadu programovat v tom hrozném jazyku.

Kouzlo Javascriptu je ale v tom, že ho má každý a bez jakékoli extra instalace. Kdo teď má počítač, má v něm prohlížeč a kdo má prohlížeč, má v něm Javascript. Nemusím se nikoho ptát, zda má potřebnou instalaci a vím, že mu Algovize na jeho počítači nebo smartphonu poběží. A vím, že Vy ho ve svém počítači nebo mobilu máte taky.

Ale ještě důležitější než výhody Javascriptu byl pro přežití Algovize důležitý jiný přístup k animacím, než měla většina animací ve zlaté době toho oboru. Je trochu obtížné vysvětlit, v čem ta “jinakost” spočívá, ale je to myslím velmi důležité, proto je to popsáno zvlášť na jiném místě této webové stránky pod názvem “Algovize - snaha o vizualizaci intuice”. Zde jen uvedu, že mi šlo a jde o to, ukázat hlavně *proč* algoritmus funguje, jaká je jeho základní myšlenka, kterou měl jeho autor v hlavě, když jej konstruoval.

Po několik let byla také na Algovizi (ze začátku Javové a pak Javascriptové) založena výběrová přednáška “Pokročilá algoritmizace” na FIT ČVUT a nyní je tam schválen výběrový předmět “Algoritmy vizuálně”.

A pak se staly ještě dvě věci, které rozvoj Algovize ovlivnily. Jak se pomalu stávám tím, čemu se nyní říká “senior”, můj kontakt s matfyzem proti mé vůli slábne. Napřed mi byl úvazek snížen na polovinu, pak na čtvrtinu, byl jsem vyškrtnut ze okruhu přednášejících ADS I/II, v roce 2020 jsem se zatím úspěšně ubránil tomu, že by moje pracovní smlouva skončila úplně. Vzhledem k velikosti (či spíše malosti) mého úvazku se další vývoj Algovize, která má nyní přes 100 000 řádků kódu, přesunul plně na nezávislou platformu algovision.org a na matfyzu se Algovize drží jen a pouze díky velkému zájmu studentů o Rozšiřující seminář ADS I a II.

A poslední událostí, která vývoj Algovize velmi ovlivnila, byl čínský virus, který začátkem roku 2020 vtrhl i do Čech a změnil do té doby přehlížené akademické debaty o distanční výuce a jejich (ne)výhodách v nejreálnější realitu.

Algovize byla vyvíjena jako pomůcka pro přednášejícího, který ví, jak ji obsluhovat a stojí před studenty a vysvětluje, co je na obrázcích vidět a co animace dělají.

Loni se ale ukázala potřeba, aby Algovize vysvětlovala na požádání sama, co ukazuje. Proto byl přidán integrovaný průvodce, který automaticky prochází celou lekcí a spouští animace a jiné grafické akce a svou činnost komentuje a vysvětluje syntetickou řečí (z technických důvodů zatím jen anglicky) a využitím anglických a českých titulků.

V současné době (březen 2021) je současná verze softwarově v zásadě hotova (i když ne zcela odladěna) a rychle přibývají průvodní texty k jednotlivým algoritmům.

A na zakončení bych chtěl vyslovit veliké poděkování studentům Rozšiřujícího semináře, jejichž připomínky, náměty a upozornění na chyby velmi přispěly k tomu, jak Algovize nyní vypadá. A doufám, že s letošními budeme pokračovat stejně úspěšně.