

Opravdoví programátoři nepoužívají Pascal

Ed Post, © 1983

Píše se rok 1983, začíná doba nástupu strukturovaného programování a programovacího jazyka Pascal, který začal vytlačovat do té doby hojně používaný Fortran či Cobol. Osobní počítač kategorie IBM PC dosud nespátřil světlo světa. Zde začíná příběh "**Opravdoví programátoři nepoužívají Pascal**", který byl publikován firmou [Datamation](#) a pojednává o tak zvaných "**Pojídačích koláčů**".

Za starých dobrých časů - ve "zlaté éře" počítačů bylo snadné odlišit muže od chlapců (v literatuře označených "**Opravdoví muži**" a "**Pojídači koláčů**"). Opravdoví muži byli ti, kteří skutečně rozuměli programování počítačů. Pojídači koláčů byli ti, co tomu nerozuměli. Opravdoví počítačový programátor říkal věci jako "DO 10 I=1,10" a "ABEND" (vyslovoval to s velkými písmeny, abyste rozuměli) a zbytek světa říkal věci jako "Počítače jsou pro mne příliš komplikované" a nebo "Nemohu se s počítačem sžít, je tak neosobní". (V předchozí práci [1] je ukázáno, že opravdoví muži nejsou vázáni na nic, nebojí se být neosobní).

Ovšem, jak je zvykem, časy se mění. Musíme se smířit se světem, v němž může získat počítač milá stará dáma ve své mikrovlnné troubě a 12-ti letý chlapec porazí opravdového muže v televizní hře na personálním počítači a personální počítač si může koupit každý, a také mu může porozumět. Opravdoví programátoři jsou ohroženi vyhynutím, existuje nebezpečí, že budou nahrazeni vysokoškolskými studenty s TRS-80.

Je zřejmé, že je potřeba ukázat rozdíl mezi typickým, vysokoškolsky vzdělaným hráčem vi-deo-her a opravdovým programátorem. Ukáže-li se tento rozdíl dostatečně jasně, dá se dětem šance, aby aspirovaly být podobní svým otcům. Pomůže to rovněž zaměstnavatelům opravdových programátorů tím, že jim to ozřejmí chybu, které se dopouštějí, když nahrazují opravdové programátory ve svém týmu mladými hráči her (a šetří tím platy).

1.0 Jazyky

Nejsnadnější způsob, jak zjistit, kdo je v davu opravdový programátor, spočívá ve zjištění, jaký programovací jazyk používá. Používá FORTRAN. Pojídači koláčů používají Pascal. Nicolaus Wirth, tvůrce Pascalu, při jedné své přednášce byl dotázán, jak vyslovovat jeho jméno. Odpověděl: "Můžete mě volat jménem (Wirth) a nebo hodnotou (Worth)". Z této poznámky je zřejmé, že N.Wirth je pojídač koláčů. Opravdový programátor uznává jediný mechanismus předávání parametrů "voláním hodnotou při návratu - tj. referencí" tak, jak je to implementováno v kompilátorech IBM/370 FORTRAN G a H. Opravdový programátor nepotřebuje ke své práci všechny tyto abstraktní pojmy, je úplně šťastný, má-li děrovač štítků, kompilátor FORTRAN 4 a pivo.

- Opravdový programátor zpracovává seznamy ve FORTRANu.
- Opravdový programátor pracuje s texty ve FORTRANu.
- Opravdový programátor řeší zpracování hromadných dat (když už to dělá) ve FORTRANu.
- Opravdový programátor řeší umělou inteligenci ve FORTRANu.

Akademici v počítačových vědách v posledních letech tvoří vyjetou kolej strukturovaného programování. Tvrdí, že programům lze snadno porozumět, když se při jejich psaní používají některé speciální konstrukce a postupy. Ne všichni se přesně shodnou na tom, které konstrukce a příklady používat, aby ukázali jejich konkrétní pohled na věc. Téměř vždy se vejdou na jedinou stránku časopisu - jsou nepřesvědčivé. Když jsem opustil školu, myslel jsem si, že jsem nejlepším

programátorem na světě. Uměl jsem psát nedostižné tic-tac-toe programy, používat 5 různých programovacích jazyků a psát programy o 1000 řádcích, které pracovaly. (Skutečně!!) Pak jsem přišel do skutečného světa. Mým prvním úkolem bylo přečíst si program ve FORTRANU, který měl 200000 řádků, porozumět mu, a pak ho 2x urychlit. Každý opravdový programátor vám řekne, že veškeré strukturované programování v celém světě vám v takovém případě nepomůže, chce to talent. Neboť:

- Opravdovým programátorům nevadí používání GOTO
- Opravdový programátor může napsat cyklus DO na 5 stran a nesplete se.
- Opravdový programátor miluje aritmetické IF, program dělá zajímavější.
- Opravdový programátor píše samomodifikující se programy, zvláště v tom případě, když může ušetřit 20 ns v nejvnitřnějším cyklu.
- Opravdový programátor nepotřebuje poznámky - vlastní kód je zřejmý.
- Poněvadž FORTRAN neobsahuje strukturované IF, REPEAT, UNITL ani CASE, opravdový programátor nezná starosti s jejich použitím.

Abstraktní datové typy, struktury, ukazatele, seznamy, řetězce - to jsou pojmy v jistých kruzích poměrně populární. Wirth (výše zmíněný pojídač koláčů) skutečně napsal celou knihu o tom, jak psát programy založené na datových strukturách (Algoritmy + Datové struktury = Programy) a ne jinak. Jak ví každý opravdový programátor, jedinou užitečnou datovou strukturou je pole. Řetězce, seznamy, struktury, množiny - to vše jsou speciální případy polí a lze s nimi tak zacházet a není nutné váš programovací jazyk komplikovat. Nejhorší věc na představě datových typů je, že je nutno je deklarovat, zatímco jazyky pro opravdové programování, jak všichni známe, mají implicitní určování typu založené na 1. znaku 6-ti znakového jména proměnné.

2.0 Operační systémy

Jaký typ operačního systému používá opravdový programátor? CP/M? Bože chraň - CP/M je přes všechny jeho klady hračičkový operační systém. CP/M mohou pochopit dokonce i postarší dámy a absolventi základní školy.

UNIX je ovšem poněkud komplikovanější - typický UNIXácký pisálek si nikdy nevzpomene, jak se tento týden volá příkaz PRINT - ale když systém má právo vypadnout z činnosti, pak je UNIX pouze glorifikovanou televizní hrou. Lidé nepracují na UNIXových systémech seriózně: posílají si vtipy kolem světa pomocí sítě UUCP a píšou dokonalé hry a vědecké články.

Ne, váš opravdový programátor používá OS/370. Dobrý programátor ve svém manuálu JCL nalezne popis chyby IJK051 a porozumí mu. Velký programátor umí psát JCL programy, aniž by se podíval do manuálu. Skutečně velmi dobrý programátor umí najít chybu i v 6MB výpisu operační paměti, a to aniž by použil hexadecimální kalkulátor (viděl jsem to skutečně dělat).

OS je skutečně znamenitý systém. Lze na něm ztratit dny práce kvůli jedné chybějící mezeře, což podněcuje programátorský tým k ostražitosti. Nejlepší způsob styku se systémem je přes děrovač dřevných štítků. Někteří lidé prohlašují, že na OS/370 běží vícepřístupový terminálový systém, ale po pečlivém studiu jsem dospěl k závěru, že se tito lidé mýlí.

3.0 Programovací prostředky

Jaký typ prostředků používá opravdový programátor? Teoreticky platí, že opravdový programátor může počítači zadávat svůj program klávesami na předním panelu počítače. V dávné minulosti, kdy

počítače měly přední panel, se to tak skutečně občas dělalo. Váš typický opravdový programátor znal celý zavlekač v hexadecimálním vyjádření zpaměti a namačkal ho kdykoliv jej váš program porušil (připomeňme si - paměť byla paměť, při vypnutí napájení jste nic nezapomněli. Dnes paměť buďto zapomene věci, které jste zapomenout nechtěli, nebo si pamatuje věci dlouho poté, kdy by bylo lépe na ně zapomenout). Legenda vypravuje, že Seymore Cray, tvůrce superpočítače Cray 1 a většiny počítačů fy Control Data, skutečně namačkal první OS do počítače CDC7600 při jeho prvním zapnutí sítě přes jeho přední panel a to zpaměti. Nutno říci, že Seymore byl opravdový programátor.

Jedním z mých favoritů na opravdového programátora byl systémový programátor od Texas Instruments. Jednoho dne ho zavolal telefonem uživatel, jehož systém se zbouřil uprostřed úklidu vysoce důležité práce. Jim byl schopný opravit poruchu po telefonu, donutil uživatele, aby si namačkal diskové V/V operace z panelu, opravil systémové tabulky v hexadecimálním vyjádření a aby mu přečetl registry zpět do telefonu. Poučení z této příhody: zatímco opravdový programátor obvykle zahrnuje "do své opravářské brašny" děrovač štítků a tiskárnu, téhož může dosáhnout telefonem a z předního panelu.

V některých společnostech už editování textu nespočívá v tom, že desítky inženýrů sedí u děrovače děrných štítků. Skutečně. V budově kde pracuji, není jediný děrovač děrných štítků. Opravdový programátor musí spojit práci v takovém případě vykonávat pomocí programu typu "text editor" ap. Většina systémů nabízí celou řadu textových editorů, opravdový programátor si musí pečlivě vybrat ten editor, který nejlépe odpovídá jeho životnímu stylu. Mnozí lidé věří, že nejlepší textové editory na světě byly napsány ve výzkumném středisku fy XEROX v Palo Altu pro jejich počítače ALTO a DORADO. Bohužel žádný opravdový programátor nikdy nepoužije počítač, jehož operační systém se jmenuje Všední hovor a určitě nebude hovořit k počítači s myší.

Některé myšlenky použité v editorech fy XEROX byly zahrnuty do editorů běžících pod mnohem vhodněji pojmenovaných operačních systémech - příkladem může být EMANCE a VI. Problém s těmito editory spočívá v tom, že opravdoví programátoři považují vlastnost "Co vidíš je to, co dostaneš" za stejně špatnou jak u textových editorů, tak i u žen. Ne, opravdový programátor si žádá editor typu "dostaneš to, co žádáš" - editor komplikovaný, tajemný, mocný, neodpouštějící, nebezpečný. Takovým je třeba TECO.

Zjistilo se, že posloupnost příkazů TECO připomíná mnohem více poruchy na komunikační lince, než čitelný text. Jednou ze vstupních her, která se dá hrát s editorem TECO spočívá v tom, že zadáte svoje jméno jako příkaz a zkusíte uhodnout, co to udělá. Zatímco budete rozprávět s TECO o chybě vstupních dat, váš program se pravděpodobně zruší nebo - což je horší - se do již odladěného programu zavlečou nenápadné a tajemné chyby.

Z tohoto důvodu opravdoví programátoři nejsou ochotni editovat program, jenž už téměř chodí. Raději udělají záplatu přímo do binárního kódu programu na disku a to pomocí čarokrásného programu zvaného SUPERZAP (nebo pomocí jeho ekvivalentu na strojích jiných než jsou stroje IBM). Ten pracuje tak dobře, že mnohé činné programy IBM systému nejsou v žádném vztahu ke svým původním FORTRANovským textům. Mnohdy není už původní zdrojový text dostupný vůbec. Když vyvstane potřeba upravit takový program, žádný manažer či šéf nebude ani chvíli uvažovat, že by na tuto práci poslal někoho jiného než opravdového programátora - kterýkoliv strukturovaný programátor typu pojídač koláčů by ani nevěděl, kde má začít.

4.0 Co opravdový programátor nepoužívá?

- Fortranské pre-procesory typu MORTRAN či RAFTOR, kuchařské umění programovat - to je něco ohromného pro dělání koláčů.

- Ladicí program na úrovni symbolického jazyka. Opravdový programátor umí číst výpisy operační paměti.
- Kompilátor provádějící kontrolu překročení mezí polí. Takové kompilátory dusí tvořivost, znemožňují většinu zajímavých použití příkazu EQUIVALENCE a znemožňují dělat změny operačního systému zadáním záporného indexu. A co je nejhorší, kontrola překročení mezí snižuje efektivnost.
- Knihovní systémy zdrojových programů. Opravdový programátor má svůj program zamčený na děrných štítcích, což mu znemožňuje ponechat svůj program nestřežený [5].

5.0 Opravdový programátor při práci

Kde typický opravdový programátor pracuje? Jaký typ programu je pro vynaložení úsilí tak talentovaných jedinců dostatečně cenný? Můžete si být jisti, že žádný opravdový programátor neumře při psaní programů pro zpracování dat v COBOLu a nebo při třídění. Opravdový programátor si žádá úkoly světové důležitosti.

- Opravdový programátor pracuje pro Národní laboratoř v Los Alamos, kde píše programy simulující atomové bomby na superpočítači Cray 1.
- Opravdový programátor pracuje pro Národní bezpečnostní úřad a dekoduje odposlouchaný radiový provoz Sovětů.
- Bylo důležité, aby tisíce opravdových programátorů pracovalo pro NASA, aby se Američané dostali na Měsíc a zpět dříve než Sověti.
- Počítače v raketoplánu byly programovány opravdovými programátory.
- Opravdoví programátoři pracují pro Boeing, kde navrhují operační systémy pro křížující rakety.

Někteří nejděsivější ze všech opravdových programátorů pracují v Kalifornii v Jet Propulsion Labs. Mnozí z nich znají celý operační systém kosmických lodí Pioneer a Voyager z paměti. S kombinací ohromných pozemních FORTRANských programů a malých assemblerovských programů v kosmické lodi jsou schopni udělat neuvěřitelné navigační a improvizace výkony - strefit se do desetikilometrového okna v případě Saturnu a to po 5-ti letech pobytu ve vesmíru, opravit nebo obejít porouchané čidlo, baterii ap. Říká se, že jeden opravdový programátor zorganizoval v lodi Voyager doplnění programu pro rozpoznání, a to do několika stovek nevyužitých bytů v paměti. Doplněk programu našel, určil umístění a vyfotografoval nový měsíc Jupitera.

Stávající plán platný pro kosmickou loď Gallileo je založen na využití pomoci gravitace Marsu při cestě k Jupiteru. Dráha této lodi prochází 80 +/-3 km od povrchu Marsu. Nikdo nebude důvěřovat programu psanému v PASCALu (nebo Pascalovskému programátoru) tak, aby mu svěřil navigaci v těchto tolerancích.

Můžeme říci, že mnozí ze světových opravdových programátorů pracují pro vládu USA - především pro ministerstvo obrany. Tak to má být. V poslední době se ovšem formuje na nebi opravdových programátorů černý mrak. Vypadá to tak, že někteří vysoce postavení pojídači koláčů v ministerstvu obrany rozhodli, že všechny obranné programy budou psány ve stejném velkolepém unifikovaném jazyku zvaném ADA ((C), DoD). Prozatím se zdá, že ADA byla orientovaná tak, aby se stala jazykem, který by bojoval proti poučkám opravdového programování - jazykem se strukturou, s datovými typy a středníky. Krátce, jazyk určený pro zmrzačení tvořivosti typického opravdového programátora.

Naštěstí jazyk přijatý ministerstvem obrany má dost zajímavých vlastností, které ho nečiní nebezpečným - je neuvěřitelně složitý, zahrnuje metody, jak si vyměňovat zprávy s operačním systémem a jak uspořádat paměť a Edgar Dijkstra ho nemiluje. (Dijkstra, jsem si jist, že ho znáte, byl autorem "GoTos Considered Harmful" - základní práce programovací metodologie, opěvované jak pascalovskými programátory, tak pojídači koláčů). Mimo to, opravdový programátor umí psát FORTRANské programy v kterémkoliv jazyce.

Opravdový programátor může přistoupit ke kompromisům ve svých principech a může pracovat i na něčem méně triviálním, než je zničení života ve formě, jak ho známe. To za předpokladu, že to přinese dost peněz. Existuje např. několik opravdových programátorů tvořících televizní hry u fy Atari. (ale nehrají je, opravdový programátor v každém okamžiku ví, jak nad strojem zvítězit - to není žádný problém). Každý kdo pracuje u fy Lucas Film, je opravdový programátor. (Bylo by ztřeštěné odmítnout peníze od 50-ti milionů fanoušků hry Star Trek). Množství opravdových programátorů pracujících v počítačové grafice je poněkud pod normou; to především proto, že dosud nikdo nenalezl pro počítačovou grafiku vážné použití. Na druhé straně, veškerá počítačová grafika je ve FORTRANu, takže existuje množství lidí, kteří raději dělají grafiku, než aby psali programy v COBOLu.

6.0 Opravdový programátor při hře

Obecně platí, že opravdový programátor si hraje stejným způsobem jako pracuje - to je s počítačem. Stále se diví, že ho jeho zaměstnavatel skutečně platí za to, aby dělal to, co by měl dělat, a to při jakékoliv legraci (i když pečlivě dbá na to, aby tento názor nevyjadřoval moc nahlas). Opravdový programátor občas opustí svůj úřad. Buď už proto, aby vyšel na čerstvý vzduch anebo na pivo či na oboje. Mezi jisté typy pro rozpoznání opravdového programátora mimo výpočetní středisko patří:

- Na večírku opravdoví programátoři v koutku hovoří o bezpečnosti operačních systémů a o tom, jak ji obejít.
- Při fotbalu opravdový programátor porovnává hru proti své simulaci natištěné na 11 až 14 stranách skládaného papíru.
- V lázních u moře kreslí opravdový programátor do písku bloková schemata.
- Při pohřbu je opravdový programátor jedinec, který říká: "Ubohý Jiří. A ten třídící program mu těsně před tím, než dostal infarkt, téměř chodil".
- V samoobsluze je opravdový programátor ten, který trvá na tom že si na pokladně namačká cenu konzerv sám, poněvadž nikdy neuvěří, že by děrovačka něco naděrovala napoprvé správně.

7.0 Přirozený domov opravdového programátora

Jaké prostředí je pro opravdového programátora nejlepší? Pro šéfa opravdových programátorů je to velmi důležitá otázka. Když uvážíme, kolik peněz stojí udržení jednoho jedince v týmu, je snažší dát ho (nebo ji) do prostředí, kde může dělat svoji práci pohodlně.

Typický opravdový programátor žije u terminálu. Mezi to, co obklopuje tento terminál patří:

- Výpisy všech programů, na kterých opravdový programátor kdy dělal, nakupené v přibližně chronologickém pořadí na každém volném místě podlahy kanceláře.
- Několik téměř či úplně plných šálek studené kávy. Občas v této kávě plave špaček nedokouřené cigarety. Někdy takový šálek obsahuje i slupky od pomeranče.

- Pokud není velmi dobrým programátorem, pak zde budou kopie manuálů JCL a Principů operací otevřené na některém zajímavém místě.
- Na stěně visí kalendář pro rok 1969 tištěný tiskárnou počítače s obrázkem Snoopyho.
- Po koberci jsou poházeny obaly od másla a plněných sýrových tyčinek (takových, které se nechají zvětrat a vyschnout ještě v obchodě, aby se už nemohly více pokazit v automatu).
- Na stole leží šablona pro kreslení blokových schemat. Tu zde nechal předchozí majitel kanceláře. (Opravdoví programátoři píší programy, ne dokumentaci. To nechávají na lidech od údržby).

Opravdový programátor je schopen pod nátlakem ve stresu pracovat 30, 40 i 50 hodin. Je fakt, že tento způsob práce preferuje. Dlouhá doba odezvy systému opravdového programátora neobtěžuje, dává mu šanci, aby si mezi kompilacemi zdřímnul. Není-li vyvinut na opravdového programátora dostatečný tlak, pak má snahu komplikovat věci tím, že pracuje stále na stejné, malé, ale zajímavé části problému prvních devět týdnů a zbytek pak dodělá za týden. To nezpůsobuje peklo pouze jeho šéfovi, který zoufá nad tím, že se projekt nedokončí, ale vytváří to i vhodnou omluvu, proč se nedělá dokumentace. Obecně platí:

- Žádný opravdový programátor nepracuje od 9 do 5 (tedy ve dne).
- Opravdový programátor může, ale nemusí znát jméno své ženy. Zná ovšem z paměti celou kódovou tabulku ASCII či EBCDIC.
- Opravdový programátor neumí vařit. Obchody s potravinami nejsou otevřeny ráno ve 3 hodiny.

8.0 Budoucnost

Jaká je budoucnost? Je v zájmu opravdových programátorů, aby poslední generace programátorů byla vychována se stejným názorem na život jako jejich starší kolegové. Mnozí z nich už ani neviděli přední panel počítače. Současní absolventi středních škol neumí řešit aritmetické operace v šestnáctkové soustavě bez vhodné kalkulačky. Současní absolventi vysokých škol jsou měkčí, jsou chráněni před programátorskou realitou ladicími systémy na úrovni zdrojového jazyka, editory, které počítají závorky a uživatelsky orientovanými operačními systémy. A co je nejhorší, mnozí z domělých počítačových vědců dělají vše pro to, aby bylo možné absolvovat vysokou školu bez znalosti FORTRANu. Jsme určeni k tomu, abychom se stali výrobci UNIXovských písálků a PASCALských programátorů?

Na základě svých zkušeností mohu říci, že budoucnost opravdových programátorů je světlá. Ani OS/370 ani FORTRAN neprokazují žádné znamení ústupu, a to navzdory veškerému úsilí PASCALských programátorů na celém světě. Ztroskotaly i mnohé záludné triky jako je přidání strukturovaných řídicích konstrukcí do FORTRANu. Samozřejmě, mnozí výrobci počítačů přicházejí s kompilátorem FORTRANu 77, ale každý takový kompilátor má způsob, jak vrátit sebe sama zpět do kompilátoru FORTRANu 68 (zadáním příkazu na štítku) a kompilovat cyklus DO tak, jak Bůh mínil, že se to dá dělat.

UNIX už dále nemůže být tak nedobrý k opravdovým programátorům jako byl. Poslední verze UNIXu je pro opravdové programátory z hlediska operačního systému cenná - nabízí dva různé a vzájemně nekompatibilní uživatelské interfacery, tajemný a komplikovaný driver dálkopisu a virtuální paměť. Když pomineme skutečnost, že UNIX je strukturovaný, programování v C mohou opravdoví programátoři ocenit: žádná kontrola typu, jména mohou mít 7 (10?, 8?) znaků a dodaný užitek datového typu

ukazatel není k zahazení - podobně jako mít na jednom místě nejlepší části jazyka FORTRAN a assembleru (a to nemluvíme o define).

Ne, budoucnost není tak špatná. Proč v posledních několika letech dokonce i populární tisk komentuje jasnou novou úrodu počítačových hrdinů a pisálků opouštějících místa jako je Stanford či MIT [7,8].

Ze všech důkazů plyne, že duch opravdového programování leží na těchto mladých mužích a ženách. Pokud existují špatně definované cíle, bizarní chyby a nereálné plány, jsou zde opravdoví programátoři, kteří přiskočí a problém vyřeší, dokumentaci nechají na později. Ať žije FORTRAN!

9.0 Literatura

[1] Feirstein, B., *Real Men Don't Eat Quiche*, N.Y., Pocket Books, 1982

[2] Wirth, N., *Algorithm + Data Structure = Programs*, Prentice Hall, 1976

[3] Xerox PARC editors

[4] Finseth, C., *Theory and Practice of Text Editor - or a cook-book for EMACS*,
N.S. Thesis, MIT/LCS/TM-165, 1980

[5] Weinberg, Q., *The Psychology of Computer Programming*, N.Y.,
Van Nostrand Reinhold, 1971

[6] Dijkstra, E., *On the GREEN language submitted to the DOD*,
Sigplan Notices, Vol3, No.10, Oct. 1978

[7] Rose, Frank, *Joy of Macking*, Science 82, Vol.3, No.9, Nov. 1982

[8] *The Hacker Papers*, Psychology Today, Aug. 1980, Oct. 1978

[9] sdcarl ! lin, "Real Programmers", UUCP-net, Thu Oct 21 16:55:16 1982