



ZMENY VO VÝUČBE UNIXU NA FEI STU OD ŠK. ROKU 2010/2011

FODREK, Peter, (SK); FOLTIN, Martin (SK); BLAHO, Michal (SK)

Abstrakt. Unix sa, pre bývalý študijný program automatizácia a dnešný študijný program Priemyselná informatika (PI), učí už približne 20 rokov. V analýze [1] prednesenej na akcii 11th Real-Time Linux Workshop (RTLWS11) konanej, pod vedením Real Time Linux Foundation Working Group neziskovej organizácie Open Source Automation Development Lab (OSADL), dňa 28.–30. septembra 2009 v Drážďanoch sme analyzovali príčiny neoblíbenosti predmetu. Existujú tri typy príčin. Jedným typom je prístup väčšiny ZŠ a SŠ vo výučbe informatiky k Linuxu a programovaniu. Druhá oblasť je nedostatok počítačov s predinštalovaným Linuxom. Tretia oblasť problémov je neprispôsobenie pedagogických metód zmeneným podmienkam. A práve túto skupinu problémov chceme riešiť pomocou prístupov uvedených v tomto príspevku.

1 Úvod

Výučba systémového paralelného programovania v Unixe, na úrovni utilít, prešla za približne 20 rokov drastickými zmenami. Učilo sa postupne na operačných systémoch a distribúciách GNU/Linux

- a) QNX
- b) Solaris 2.0
- c) Mandrake Linux 9.2 +Debian
- d) Fedora Core 4 +Debian

Dvaja starší autori príspevku, v roku 1997, zažili, ako študenti, Solaris 2.0 s Open Windows na dvoch X-termináloch a telnetový prístup z MS-DOSu na solarisový server z ďalších 8 počítačov v dvoch učebniach. Študenti sa striedali tak, že jeden týždeň boli v jednom a druhý v inom laboratóriu. V každom laboratóriu bol prítomný iný z dvoch cvičiacich,

ktorý si miesto nemenili pri výmene študentov. To dávalo možnosť porovnávať ako aj prijať rôzne prístupy k problému. Navyše bol predmet v 3. nominálnom ročníku štvorročného bakalárskeho štúdia a predchádzal mu predmet Operačné systémy.

Na Operačných systémoch (OS) sa učilo skriptovanie v tesh na súborových systémoch so zákernosťami: ako súbor s nula až troma medzerami alebo netlačiteľnými znakmi na začiatku mena súboru, či symbolický link na vyššiu úroveň v strome s tým, že sa hodnotilo stratou 10 z 15 bodov to, že sa skript nezacyklil. Posledné zadanie z OS bolo na medziprocesovú komunikáciu (IPC) a procesy v jazyku C. OS sa učil v letnom semestri 2. nominálneho ročníka ako povinne voliteľný predmet. Keďže však bol jediným povinne voliteľným predmetom a študenti si museli zvoliť najmenej jeden povinne voliteľný predmet z ponuky, bol de facto povinným. Potom, ako sme OS absolvovali, sa predmet OS začal hodnotiť tak, že namiesto dvoch týždňov sa muselo vypracovať zadanie za dve 50-minútové hodiny. To viedlo k tomu, že v priebehu dvoch rokov získali zápočet len 4 študenti nášho študentského odboru (vtedy ešte). Pritom ročníky mali 120 a 100 študentov. Keďže predmet učila vtedajšia Katedra informatiky a výpočtovej techniky (dnešná FIIT) vytvorila sa „alternatíva“ vo forme neUnixového predmetu. Toto považovali KIVT za podraz a urážku, čím sa začal proces delenia fakulty na FEI a FIIT.

Je to smutné, lebo z teoretických znalostí z OS sa dalo ťažiť dodnes. Len vďaka nim bolo možné pochopiť nové algoritmy plánovania procesov pre OS reálneho času, ktoré boli prednesené na RTLWS11 v Drážďanoch.

Faktické zrušenie predmetu OS navyše zásadne zmenilo podmienky výučby predmetu. Hoci na OS mal každý študent len jednu konkrétnu formu IPC a na vtedajšom predmete Programovacie prostriedky reálneho času sa preberali všetky formy a prostredky na IPC, knižnica nCurses a sockets, čo je však len špeciálna forma IPC, bol predmet OS veľkou pomocou. Našou výhodou bolo aj to, že ak sme chceli používať Internet a nemali sme iný prístup k nemu, museli sme od prvého ročníka používať prístup cez telnet z MS-DOSu na server DECEF (z DEC na ElektroFakulte) prípadne k DECEFu pristupovať pomocou textových terminálov vdt52s. Prehliadače boli lynx a www. Na DECEFe bežal Ultrix t. j. Unix od Digitalu (DEC = Digital Equipment Corporation), neskôr FreeBSD, openBSD a netBSD. Predmet OS sa učil na vdt52s na serveri ALF (podľa procesoru DEC Alpha AXP). „Na ALFovi“ bol okrem prehliadačov www a lynx aj NCSA Mosaic, ktorý sa dal spustiť na X-termináloch. DECEF bežal na architektúre VAX, dnes je aj on PC-serverom s BSD.

Najmladší autor Ing. Blaho, absolvoval predmet na Mandrake (po zlúčení s Connectivou sa dnes volá Mandriva) so serverom KAR10 na Debiane. Spoluautor Fodrek učí dnes predmet na Fedore. Ak si pripomnieme, že Výpočtové stredisko FEI, a s ním aj Centrálna počítačová učebňa bežia na Windows 7 a učebňa na predmet má dualboot s Windows XP, vychádza jasne, že treba zmeniť prístup k výučbe. Navyše LDAP, NFS a ssh server KAR10 bol vymenený za KAR11. Stále sa však toleruje prístup na KAR11 (KAR je skratka bývalej Katedry automatizácie a regulácie, ktorá je po zlúčení súčasťou Ústavu riadenia a priemyselnej informatiky) cez ssh a priamo sa študentom ponúka Windows based ssh klient Putty.

2 Aktuálny stav výučby

Napriek uvedeným zmenám v pripravenosti študentov na štúdium predmetu nedošlo k zmene filozofie a prevedenia predmetu. Skôr naopak. Pri zmene bakalárskeho štúdia sa skrátili cvičenia z troch na dve hodiny týždenne. Nedôveryhodne pôsobia dnešným študentom aj prednášky na priesvitkách ukazovaných cez spätný projektor. Vrcholom ale je, že vedúci cvičení, ktorý nie je autorom, prichádza občas na cvičenia s notebookom s Windows. Pritom študentom hovorí, aký je Unix a Linux dobrý.

Problém vypukol aj, keď sa predmet učil jeden rok aj pre študentov štúdiijného programu Aplikovaná informatika (AI). Úspešnosť študentov v získavaní zápočtov AI bola 44% a PI bola 88%. Aj za tých 88% sme dostali ústne napomenutie od vtedajšieho riaditeľa ústavu spolu s príkazom zvýšiť priechodnosť predmetu. Študenti AI už na ďalší rok nemali v programe tento predmet. Práve 44% je podľa Saeda Dehnadiho [2] a iných štúdií percento študentov IT schopných naučiť sa programovať.

Keďže študenti AI, ktorí nezískali zápočet sa nedostavili ani na riadny ani na opravný zápočet, ktorý spočíval v odovzdaní 3. zadania, nemohli sme konať inak. Zarážajúci, ale bol rozdiel medzi študentami programu, kde bol predmet nový, a programu, kde už poznali zadania daného typu. V reakcii na to sme zmenili zadania na také, že sa neopakujú a ani nemôžu. Ide o simuláciu sieťových regulačných obvodov, kde na cca. 62 študentov je 13 regulačných obvodov. 12 z regulačných obvodov má po dva riadené systémy a pre každý typ riadených systémov je bežne 6 rôznych prenosových funkcií najmenej však 3 prenosové funkcie. To dáva od 118 do 438 rôznych zadaní. Stačí však vygenerovať každých 4-5 rokov nové prenosové funkcie a zadania sa nebudú opakovať. Predtým to bolo 15 rôznych technologických procesov, ktoré však nemali daný matematický opis a tak boli stále rovnaké.

Výsledkom tejto zmeny bol inzerát v internej sieti študentského domova, ktorý ponúkal po 2 000 Sk za vypracovanie každého z dvoch kontrolných zadaní. Tieto zadania sa odovzdávajú v 5. a 9. týždni semestra a majú hodnotu po 10b. Navyiac tam bola ponuka 5 000 Sk za záverečné zadanie za 30 bodov. Na inzerát nikto nereagoval, lebo sa študentom, podľa výrazu dvoch najkvalitnejších študentov v ročníku, neoplatilo za také peniaze vypracovať zadanie kolegom. Zaujímavé je, že študent, ktorý sa priznal k autorstvu inzerátu, vypracoval nakoniec zadania sám na 46 z 50 možných bodov. Ale na záver podotkol, že mu to zabralo príliš veľa času.

O rok sme teda porovnávali časy potrebné na vypracovanie zadaní s časmi uvedenými v štúdiijnom programe. Platí totiž pravidlo, že počet kreditov za predmet sa rovná súčtu týždenného počtu hodín prednášok, cvičení a samostatnej domácej práce resp. prípravy na predmet. Keďže predmet má 5 kreditov, 2 hodiny prednášok a 2 hodiny cvičení, má byť teda 1 hodina domácej prípravy týždenne. Ako to ale vyzerá v skutočnosti? Študenti uvádzajú, že pri riešení prvého zadania doma strávili 15–18 hodín. Pri druhom zadání to bolo 14–15 hodín a pri treťom 40–45 hodín. To dáva dohromady 70–80 hodín domácej prípravy za semester. To je však obrovský nepomer k plánovanej jednej hodine týždenne, teda 12–13 hodinám za semester. Naša veková kategória mala iba jedno 50 bodové zadanie,

ale bolo odovzdávané od 8. týždňa, pričom najlepší jedinci nemali dva týždne pred minimálnym týždňom odovzdávania čo robiť. A to si ešte sami museli najskôr do 3. týždňa vymyslieť zadanie, aby tam boli všetky preberané oblasti, a ktoré nakoniec odobril cvičiaci. Na riešenie sme teda mali 3. týždne a celkovo riešenie zabralo 25–30 hodín (pol dňa týždenne v prestávkach vo vyučovaní a pol soboty alebo nedele doma).

Vtedy, keď sme prieskum robili, tak mal Ing. Fodrek názor, že obsah predmetu pokrýva len 50% reálnych potrieb programátora systémov reálneho času. Po RTLWS11 ho upravil tak, že predmet pokrýva len 25% potrebných vedomostí. Pri časovej náročnosti predmetu je to však neriešiteľný problém.

3 Realizované zmeny

Ostáva aplikovať zmeny vo výučbe. Dve už nastali, a hoci problém len zvýraznia, majú pozitívny vplyv na vzdelanosť. Prvou je rozdelenie dvojíc na jednotlivcov. Tým sa časová náročnosť de facto zdvojnásobí. Aj keď sa vie, že aj tak dominantnú časť, alebo celé zadanie, vyriešil jeden študent a druhý sa mu odvdáčil vypracovaným zadaním z iného predmetu. Druhá je zmena názvu. Po Programovacích prostriedkov reálneho času máme dnes Sotvér riadiacich systémov I. Od šk. roku 2010/2011 sa bude predmet volať Unix/Linux – systémy reálneho času. Zároveň sa zmenila dotácia na 3 hodiny cvičení a zostali dve hodiny prednášok ako aj hodina predpokladanej domácej práce. Z toho vyplýva pridanie kreditu. Predmet sa presunul z letného do zimného semestra II. nominálneho ročníka. V súvislosti s odchodom prednášajúcej do dôchodku ku 1. 11. 2010 sa mení aj prednášajúci. Podľa štúdiijného programu by ním mal byť terajší bežný cvičiaci Fodrek.

4 Zamietnuté zmeny

Existoval návrh na spojenie predmetov Základy systémov reálneho času (ZSRT), ktorý sa premenuje na Java – grafické rozhranie RS. Keďže pôvodne bolo ZSRT náhradou za Operačné systémy, bolo by to logické. Navyše by to umožnilo presunúť výučbu Javy z Windows na GNU/Linux. Tak isto by bolo možné mať jediný vývojový prostriedok na oba predmety napr. Eclipse. Eclipse ovláda ako C používané v Unix predmete, tak aj Javu. Navyše Java predmet mal dotáciu 2 hodiny prednášok a 3 hodiny cvičení (po novom má 2 hodiny cvičení). Po spojení by vznikol predmet so 4 hodinami prednášok a 5 hodinami cvičení týždenne. Zmena bola navrhnutá spolu s pridaním hodín samostatnej práce na tri. To sa zamietlo, a dokonca by sa znížil počet hodín samostatnej práce na jedinú hodinu, z dvoch hodín na oba predmety v súčte. Navyše po novom je Java – grafické rozhranie RS povinne voliteľným predmetom, kde sa volí jeden z dvoch.

Z dôvodu nedostatku financií a odmietnutia grantu KEGA nemohlo byť laboratórium určené na cvičenia renovované po stránke hardvéru a ani softvéru hoci boli začaté predbežné rokovanie a akademickej licencií Red Hat Enterprise Linux MRG Realtime.

5 Navrhované zmeny v prístupe k prednáškam

V prístupe k prednáškam sa navrhuje použiť notebook s Unixom. Je to dokonca podmienka vedenia ústavu pre pokračovanie predmetu. Problémom je aký notebook vybrať? Na našom trhu by malo byť niekoľko notebookov s Linux-om ako OEM operačným systémom. Ich rozvrstvenie v podstate vystihuje pozíciu Linuxu všeobecne. Bezplatné Linuxy v najnižšej triede a platené Linuxy v najťažších kalibroch majú silnú pozíciu, ale v bežnom segmente prakticky neexistujú. Väčšina dodávateľov počítačov v SR neponúka počítače s Unixom, majú len Windows, maximálne ak ponúkajú PC bez Operačného systému.

Táto zmena spôsobí zmenu teoretických prednášok do praktickejšej formy s ukázkami konkrétnych programov, ktoré sa doteraz robili na cvičeniach. Umožní sa tým využiť lepšie čas cvičení. Nejaká miera teórie je ale nutná. Je potrebné nájsť vhodný pomer. Momentálne je to 3:1 v prospech teórie. Plán je zmeniť to na 1:1 až 1:2 v jej neprospech. Navrhuje sa aj sprístupniť prednášky mimo laboratória. To však predpokladá ochranu voči plagiátorstvu zo strany STU. Terajší stav je taký, že materiály sú dostupné len v laboratóriu cvičení. Je to preto, lebo nastal prípad, že sa na inej univerzite objavila doslovná kópia prednášok terajšej prednášajúcej, len so zmeneným menom prednášajúceho. Vedenie však odmietlo ísť do právneho sporu a tým chrániť autorské práva k prednáškam.

6 Navrhované zmeny na cvičeniach

Na cvičeniach sa navrhuje prebrať zadania každé cvičenie resp. na 10 cvičeniach po 5 bodov, alebo každý týždeň po 2 body a 30 bodov za záverečné zadanie. Tak isto sa navrhuje, aby sa obmedzili, ak nie zrušili, prezentácie demonštračných príkladov na cvičeniach. Tak budú k dispozícii len zadania jednotlivých cvičení a odpadne podobný problém s autorským právom ako bol na prednáškach.

7 Zmeny v obsahu predmetu

Vzhľadom na to, že dnes sa „čaká“ na ukončenie témy procesy na prednáškach a až potom sa pristúpi k jej preberaniu na cvičeniach, druhé a tretie cvičenie sa preberala knižnica nCurses (new curses=nové zaklínadlá, nové kliatby, nové prekliatia). Vzhľadom na snahu, aby nás, na konci semestra, študenti opäť nepreklínali, tak túto časť plánujeme zrušiť, prípadne presunúť na koniec semestra v menšom rozsahu.

Uvoľnené miesto plánujeme využiť na cvičenie obsluhy systému, ktoré sa teraz cvičí len v prvom týždni semestra. Prípadne môže byť súčasťou tejto kapitoly aj InstallFEST. Namiesto asi 1/3 prvej prednášky a jedného cvičenia by sa obsluha učila 1-2 prednášky a 1-2 cvičenia. Téma procesy sa bude prednášať dva namiesto terajších skoro troch týždňov. Cvičenia ostanú v dvojtýždňovom rozsahu. Túto tému totiž zvládajú všetci študenti. Téma

signály ostane na jednotýždňovej dotácii na prednáškach a cvičeniach, hoci sú v nej drobné nedostatky, z pohľadu koncových znalostí študentov.

Téma časovače a čas spôsobuje študentom najväčšie problémy z celého predmetu, aj preto, že sa učila v 5. týždni, keď sa preberalo zadanie prvého bloku. Preto ju navrhujeme rozšíriť na dva týždne. Ide totiž o kľúčovú tému predmetu. Téma však využíva procesy aj signály. Časovače oznamujú svoje vypršanie štandardne pomocou signálov. Sockety a spolupráca s DNS, ktoré ako-tak zvládli všetci študenti sa budú učiť rovnako ako teraz dva týždne. Ide o tretiu najdôležitejšiu tému predmetu. Vlákna robia len drobné problémy, tak ostanú na týždňovej dotácii. Možno sa presunú pred procesy alebo signály, keďže sú procesom podobné a signály im vedia byť doručené.

Nasleduje medziprocesová a medzivláknová komunikácia a synchronizácia (rady správ, rúry/pipe, zdieľaná pamäť, semaforey ...). Ide o druhú najdôležitejšiu tému. Preto jej venujeme zvyšok semestra. V rámci obsluhy sa budeme po novom venovať ladiacemu nástroju gdb. Viac sa budeme venovať gcc/g++/ln a tvorbe Makefile. Tak isto sa sem presunie dodatková téma o tvorbe binárneho spúšťačieho súboru z viacerých zdrojových kódov. Toto totiž kód sprehľadní, rovnako ako delenie kódu do funkcií, ktorému sa študenti bránia.

8 Podporné aktivity

Pokúsili sme sa otvoriť krúžok obsluhy Unixu pre stredné školy. Individuálne sa prihlásila jedna študentka a hromadne prišla 13-členná skupina tretiakov zo súkromného gymnázia. Priviedol ju ich učiteľ, absolvent FMFI UK. Z tej skupiny prejavilo záujem študovať na technickej škole asi 10 študentov, z nich len traja mali záujem o krúžok. Krúžok teda stroskotal a pravdepodobne sa nebude už konať.

9 Styk s praxou

Súborom neriešiteľných problémov je dať do súladu požiadavky trhu práce a mentálne možnosti študentov. Pán Klaus nedávno hľadal programátora v C pod Unixom na embedded systémy [3]. Podmnožinou embedded systémov sú hard realtime systémy, ktoré by mali vedieť tvoriť absolventi Priemyselnej informatiky. Uvedme si teda otázky prijímacieho testu pána Klauza, ku ktorým som pridal komentár kedy sa tá-ktorá časť učí na našom predmete.

Znalostní test

- **Jaké znáte mechanizmy meziprocesní komunikace v Linuxu a stručně je popište i s případnými klady a zápory.** (obsah otázky sa učí 1/5 I. bloku, dve tretiny II. bloku a celý III. blok)
- **Jaké znáte nástroje pro distribuovanou (různé instance OS) meziprocesní komunikaci a stručně je popište i s případnými klady a zápory?** (v celom druhom bloku a na poslednej prednáške, ktorá nie je obsahom cvičení)

- **Co je asynchronní komunikace (v rámci IPC) a kdy je výhodná?** (v 1/5 prvního bloku, v 1/6 druhého bloku a v 1/3 třetího bloku)
- **Potřebujete, aby vašich 5 aplikací mezi sebou v rámci jednoho OS a jednoho uživatele komunikovaly - předávaly si krátké zprávy. Mezi jakými topologiemi komunikace můžete vybírat a jaké jsou jejich klady a zápory.** (v 2/5 I. bloku, v celom II. bloku a v celom III. bloku)
- **Napište krátký program, který pustí najednou 5 podprocesů. Všechny podprocesy „usnou“ náhodně na 5-10 sekund a bez další akce se ukončí. Rodič čeká na všechny procesy a jakmile se jeden potomek ukončí, vypíše na stdout číslo procesu, který skončil. Pokud skončily všechny procesy, automaticky se ukončí také. K jakému problému může dojít a jak ho budete řešit?** (v 2/5 prvního bloku a v 1/3 druhého bloku)
- **Pokud spustíte program z předchozího bodu a po vzniku podprocesů zmáčknete ctrl-c, co se stane s těmito podprocesy? A proč? Jak byste změnil program, abyste změnil toto chování?** (v 1/5 prvního bloku)
- **Píšete aplikaci, ve které potřebujete zpracovávat data každou 1ms. Napište tedy smyčku, která bude číst stdin a tato data vypisovat na stdout průměrnou rychlostí jeden znak za 1ms bez ohledu na rychlost dat na vstupu. Popište všechny možné problémy, které musíte řešit, jejich původ a jejich řešení.** (v 3/5 prvního a 1/3 druhého bloku)
- **Píšete grafickou aplikaci v Linuxu, jejíž součástí má být oblast okna, které bude co nejvěrněji emulovat příkazovou řádku - terminál v rozlišení 80 × 25. Jakou rychlou techniku byste použili a na co je potřeba dát pozor?** (Táto oblasť sa neučí pre jej náročnosť. Ale jedno z možných riešení matne súvisí s nCurses, ktoré sa učia na druhom a treťom cvičení prvního bloku. nCurses ale chceme „obetovať“ v prospech obsluhy systému.)
- **Co je window manager a jaká je jeho úloha? Pomůže vám v tzv. kiosk módu aplikace např. web browser? Proč?** (Táto oblasť sa neučí, nepriamo sa spomenie jednou vetou ako voľba gdm. Korektný KIOSK mód (t.j. bez Window managera) je oriešok aj pre kóderov jadra, ktorí sú našimi absolventami.)
- **Potřebujete monitorovat chování vaší aplikace tj. vsunete do ní posílání zpráv v místech, které potřebujete sledovat. Navrhněte, jak tuto aplikaci sledovat na stovkách stanic v lokální síti? Pokud navrhnete více možností, vysvětlíte jejich klady a zápory.** (Táto oblasť sa korektno nepreberá. Využit' pri, otázkou, navrhovanom nekorektnom riešení sa dá druhý blok, 1/3 třetího bloku a obsah prednášky, ktorá sa nepreberá na cvičení.)

- **Vytvorili ste web browser postavený na existujúcej knižnici. Pri spustení sa ale vizuálne nenačítá žiadna stránka. Co byste zkusili zjistit jako první a jak?** (Toto sa neučí, iba ak okrajovo v druhom bloku a na inom predmete.)
- **Potřebujete zjistit, zda v poslední hodině vaše aplikace na některé ze stovek stanic vygenerovala chybovou hlášku. Jak to zjistíte a jaké nástroje k tomu použijete?** (Táto téma sa, de facto, neučí, lebo je o expertnom ovládaní povelového riadku a aj po pridaní času bude v predmete len rýchlokurz povelového riadku.)
- **Potřebujete, aby se vaše aplikace resp. sada aplikací spouštěla automaticky po nastartování Linuxu (kiosk mód - tj. uživatel pracuje jen s vaší aplikací). Jak to zajistíte? Je to spojeno s nějakým omezením či rizikem?** (Toto niečo medzi témou Window manager a predchádzajúcou otázkou. Preto sa to neučí. Niečo sa preberie po zmene.)
- **Zjistili jste, že vaše aplikace v produkčním nasazení je v ojedinělých případech (jednou za dva dny) nestabilní a padá. Odhadujete, že vyřešení bude vyžadovat minimálně týden práce. Co uděláte, jako rychlou a dočasnou úpravu?** (Učiť toto neprichádza do úvahy. Veď študenti nevedia dobre ani len analýzu kódu. Syn-téza je neskonálny problém. Ešte ako-tak zvládnu implemetáciu s využitím demo príkladov. A o zvyšku životného cyklu programu, okrem dvoch zo 65 študentov, ani len nepočuli)
- **Potřebujete v knihovně webkit povolit 3D transformace, které fungují jen v Safari verzi a v ostatních jsou zakázané. Zkuste lehce odhadnout, jak budete postupovat.** (Táto téma sa čiastočne učí na inom predmete.)

Ako vidno z testu je jeden predmet príliš málo na obsiahnutie problematiky. Úvodom do ťažších oblastí je bakalárska práca Michala Praženku: „Analýza plánovačov v Linuxe“ z mája 2010. Tento dištančný študent sa, ako prvý u nás, a asi aj na svete, aspoň okrajovo, zaoberal vplyvom I/O plánovačov na hard realtime. Doterajšie práce sa zaoberali len plánovačmi procesov. A pritom vstupno-výstupné operácie sú pre deadline rizikovejšie. Plánovanie procesov je dôležité pre akademikov pre obmedzenie latencie. Praktikov zaujíma deadline teda latencia+ čas vykonania úlohy. A v tej druhej časti je plánovanie I/O kritické. Aj napriek problémom sa teda občas objaví dobrý študent, ktorý vytvorí kvalitnú záverečnú prácu. Pred pánom Praženkom bol podobný Vladimír Chren, ktorý urobil, na prelome rokov 2007–8, za 45 dní kernelový port CANOpen protokolu. O to isté sa pokúšali francúzi od roku 2002 do roku 2005 a neuspeli. Globálne tu uspel koncern VW, ktorý vyvíjal CAN protokoly a ovládače od roku 2002 a až v marci 2008 mal funkčný kód. CANOpen od VW fungovalo plnohodnotne vo vanila jadre až v septembri 2009. Prítom VW v januári 2010 predstavilo prvé plne linuxové auto na svete: Audi A8 model roku 2011. Ostatné značky koncernu budú plne linuxové do konca roku 2014. BMW majú zatiaľ Linux len na audiosystéme.

Problémom je, že takých študentov je minimum a prax a predmet sú stavané na ich schopnosti. Toto potvrdzuje aj jedna z popredných vývojárov Google, absolventka Žilinskej univerzity, Petra Popluhárová. V rozhovore [4] sa vyjadrila jasne o prijímaní do Google:

„Bola to poriadna zaberačka. Nikoho nezaujímalo, aký mám titul alebo vzdelanie, ale dostávala som záludné otázky z programovania. Musela som riešiť algoritmy a rôzne hlavolamy, podľa ktorých softvéroví inžinieri zisťujú, ako to uchádzačom o prácu myslí.“

A to treba povedať, že Google sídli v oblasti Silicon Valley (medzi južným San Franciscocom a San José). Priamo v Silicon Valley, alebo do 20 km okolo neho sídli centrála, alebo IT pobočka 2.,4. a 5. najlepšej univerzity sveta v technických vedách podľa Academic Ranking of World Universities z čínskeho Šanghaja. V programovaní a IT sú to dokonca 1., 3. a 4. najlepšia univerzita. Ide o miestnu Stanfordovu univerzitu (Leeland Stanford junior University) a University of California at Berkeley ako aj Pittsburskú Carnegie Mellon University s IT pobočkou Carnegie Mellon Silicon Valley. V programovaní sa medzi uvedené univerzity vklienila len celkovo najlepšia technika: MIT (Massachusetts Institute of Technology) zo 105-tisícovej „Bostonskej Petržalky“ Cambridge, MA. Napriek tomu nemá vzdelanie a ani ukončená škola žiadny vplyv na prijímanie do práce v oblasti nami vyučovaného predmetu.

Cambridge, Massachusetts by mal byť vzorom pre vzdelávacie systémy. Na 105 tisíc ľudí má 14 univerzít. 11 z nich sú college, teda naše odborné vysoké školy. Tie môžu vydávať len bakalárske diplomy. Jedna zo 14 je obyčajnou univerzitou a dve sú výskumné univerzity. Tie majú na jedného študenta bakalárskeho štúdia troch študentov inžinierskeho/magisterského štúdia. Na jedného študenta druhého stupňa pripadá na výskumných univerzitách päť doktorandov. Najväčším problémom predmetu je, že je stavaný na uvedený model výskumnej univerzity, čo je u nás neexistujúca kategória univerzít. Navyše predmet nerešpektuje mentálne danosti študentov a ich nedostatočnú prípravu na nižších stupňoch školského systému. Vieme rešpektovať len požiadavky zamestnávateľov. To je ako regulácia s neznámym obmedzením akčného zásahu. Pokúsime sa to zmeniť.

10 Záver

Navrhli sme zmeny s ohľadom na skvalitnenie výuky predmetu pre nepripravených študentov. Napriek problémom dúfame v úspešnú implementáciu. V tomto momente sa už tvoria prezentácie na predmet, ktorý sa začne vyučovať v septembri.

Literatúra

- [1] Fodrek, P.: *Problems regarding teaching Linux programming for Industrial realtime programmers* In: Eleventh Realtime Linux Workshop+In:Linux Weekly News[online]. Dráždany+Boulder, Colorado : TU Dresden,OSADL+Eklektix, Inc. 2009, do-

stupné na <http://lwn.net/images/conf/rtlws11/papers/proc/p12.pdf>, citovane 10. 5. 2010

- [2] Dehnadi, S. – Bornat, R.: *The camel has two humps (working title)*[online]. Londýn : Middlesex University,. 2006, dostupné na <http://www.eis.mdx.ac.uk/research/PhDArea/saeed/paper1.pdf> citovane 10. 5. 2010
- [3] Klaus, J.: *Znalostní test – programátor C*[online]. Mountain View,California: Posterous(Y Combinator), 2010, dostupné na <http://jaroslavklaus.posterous.com/znalostni-test-programator-c> citovane 14. 5. 2010
- [4] Vilhanová, D.: *Slovenská googláčka* In Emma [online]. Bratislava: Emma online (Plus7, s.r.o.), 2008, dostupné na <http://emma.pluska.sk/emma/clanky/kariera-a-peniaze/uspesne-zeny/slovenska-googlacka.html> citovane 14. 5. 2010

Kontaktná adresa

Peter FODREK (Ing. , PhD.),

výskumný pracovník, Oddelenie Informačných a komunikačných technológií, Ústav riadenia a priemyselnej informatiky. FEI STU v Bratislave, Ilkovičova 3,
812 19 Bratislava, peter.fodrek@stuba.sk

Martin FOLTIN (Ing. , PhD.),

vedúci, Oddelenie Informačných a komunikačných technológií, Ústav riadenia a priemyselnej informatiky. FEI STU v Bratislave, Ilkovičova 3,
812 19 Bratislava, martin.foltin@stuba.sk

Michal Blaho (Ing.),

pedagogický pracovník, Oddelenie Informačných a komunikačných technológií, Ústav riadenia a priemyselnej informatiky. FEI STU v Bratislave, Ilkovičova 3,
812 19 Bratislava, michal.blaho@stuba.sk

Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach

1.–4. júla 2010, Žilina, Slovensko

Organizátori: Miloš Šrámek, Spoločnosť pre otvorené informačné technológie
Tatiana Šrámková, Katedra fyziky, FEI STU Bratislava
Michal Kaukič, Aleš Kozubík, Tomáš Majer, Žilinská univerzita
Lýdia Gábrisová, Ľubica Micháľková, Žilinská univerzita
Juraj Bednár, Digmia, Slovensko
Miloslav Ofúkaný, GeoCommunity, Slovensko
Peter Mráz, Kremnica
Slavko Fedorik, SOŠ elektrotechnická, Poprad
Peter Štrba, Spojená škola/Gymnázium M. Galandu, Turčianske Teplice
Ladislav Ševčovič, FEI, Technická univerzita v Košiciach

Editori: Michal Kaukič
Miloš Šrámek
Slavko Fedorik
Ladislav Ševčovič

Recenzenti: Mgr. Juraj Bednár
Mgr. Rudolf Blaško, PhD.
RNDr. Ján Buša, CSc.
Ing. Slavko Fedorik
Ing. Karol Grondžák, PhD.
Mgr. Michal Kaukič, CSc.
Ing. Tomáš Kliment
RNDr. Aleš Kozubík, PhD.
Mgr. Juraj Michálek
doc. RNDr. Štefan Peško, CSc.
Ing. Pavel Stříž, PhD.
RNDr. Ladislav Ševčovič
Ing. Michal Žarnay, PhD.

Vydavateľ: Spoločnosť pre otvorené informačné technológie – SOIT, Bratislava

ISBN 978-80-970457-0-8

Sadzba programom pdfT_EX Ladislav Ševčovič

Copyright © 2010 autori príspevkov. Príspevky neprešli redakčnou ani jazykovou úpravou.

Ktokoľvek má dovolenie vyhotoviť alebo distribuovať doslovný opis tohoto dokumentu alebo jeho časti akýmkoľvek médiom za predpokladu, že bude zachované oznámenie o copyrighte a o tom, že distribútor príjemcovi poskytuje povolenie na ďalšie šírenie, a to v rovnakej podobe, akú má toto oznámenie.