

Odvaha nebude stačiť

Programovanie je malá metafora stvorenia.

Na začiatku je objednávka, ktorá temer nikdy nie je príliš presná. Nakoniec, kto už vie do absolútneho detailu presne popísať, ako sa počítajú výplaty nepoistených práceneschopných, analyzujú otlaky prstov drevorubačov, objednávajú lieky cez Vianoce? Ak ste niekedy počúvali odborníka na dovolenke, ako sa snaží po telefóne vysvetliť svojmu nekompetentnému kolegovi riešenie neštandardnej situácie, viete, že kľúč k ozajstnému majstrovstvu spočíva v tajomnej schopnosti nazývanej improvizácia.

To sa naprogramovať nedá. Nám programátorom neostáva nič iného, ako pracne, nudne a neinvenčne popísať všetky možné situácie a detailný postup ich riešenia. Počítač je horší ako pes. Doslova poslúcha program. Ak by mal stokrát zopakovať tú istú chybu, urobí ju. Tvorba programu spočíva v detailnom analyzovaní problému, v jeho rozložení na menšie časti, naprogramovanie týchto častí, ich spojenie, testovanie, ukázanie zadávateľovi, ktorý zistí, že to malo fungovať inak. Analyzovanie zmeneného zadania, prispôbenie už existujúcich častí, doprogramovanie ostatného, integrácia, testovanie, ukázanie zadávateľovi. Ten už je vcelku spokojný s fungovaním programu, avšak pri skúšaní príde na to, že pri pôvodnom aj revidovanom zadaní zabudol na dôležitý detail, ktorý je z hľadiska jeho potrieb kľúčový. Detail sa ukáže byť nielen zložitý, ale aj meniaci logiku celého programu. Detailný popis nového zadania, výber použiteľných častí, doprogramovanie nového, integrácia, testovanie, ukázanie používateľovi. Ten už je viac menej spokojný, len ešte potrebuje vypisovať najrôznejšie formuláre, nepáči sa mu komunikácia s programom, nerozumie náповedám.

To už ale horia termíny, všetko sa rýchlo dokončuje, píšú sa manuály. Popritom sa zistí, že niekoľkonásobná úprava programu spôsobila, že prestalo fungovať aj to, čo pôvodne fungovalo, že za istých okolností (pre istotu vždy iných) program padá, odmieta najst' riešenie alebo ponúka riešenia očividne nesprávne. Nastáva fáza hektického hľadania chýb, krvavých očí po prebdených nociach, známy efekt ranného spasenia, keď chyba, ktorá vzdorovala zúfalému úsiliu o odhalenie, sa po pár hodinách spánku bez námahy objaví.

Nakoniec sa program odovzdá. Programátori sa zase vyspia, zákazník je trochu nahnevaný a trochu spokojný. Všetci vedia, že to nie je celkom ono, že to ešte nie je hotové a nie vždy to funguje. Málokedy ale ide o život či o atómovú elektrárňu a ľudia sú aj tak zvyknutí, že všade sa robia chyby.

Možno to uráža absolútne ambície, ale vedomie omylnosti je nevyhnutná výbava rozumného programátora. Pri programovaní čo i len mierne zložitého problému je desivo veľa možností. Stačí raz zabudnúť a program, neschopný sebatriviálnejšej reflexie, spadne do pasce a zrúti sa. Nie je v ľudských silách dopredu všetko premyslieť, otestovať všetky alternatívy. Pre programátora je chyba (alebo dôvernejšie bug) urážkou, nočnou morou a každodenným verným spoločníkom. Kvôli nim nikdy nevieme, kedy budeme hotoví. Kvôli nim típneme, keď si k novému programu sadá používateľ. To sú naše malé infarkty. Kto s nimi nedokáže žiť, musí si najst' iné remeslo.

Programovanie je malá metafora stvorenia. A teraz si predstavte, že odvážni chlapci sa chystajú opravovať genetický kód. V najdokonalejšom programe neznámeho majstra, napísanom v jazyku, ktorému celkom nerozumieme, na základe zadania, o ktorom nič nevieme, opravovanom veky, chcú odstraňovať chyby, meniť vlastnosti, pridávať nové funkcie. Žasnem nad ich trúfalosťou. Som zvedavý, ako budú debuggovať.