

# Open Source Software

STANISLAVA KEKEROVÁ

*Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 3, 842 16 Bratislava*

**Abstrakt.** Účelom tejto eseje je priblížiť a vysvetliť potencionálnemu čitateľovi, čo sa vlastne skrýva pod pojmom Open Source Software (OSS) – voľne sa dá tento názov preložiť ako voľne šíriteľný softvér. Opíšem jeho vznik, trochu málo priblížim tvorcov tohto softvéru, zodpovedám otázku prečo vlastne ľudia investujú svoj voľný čas na jeho tvorbu.

Keďže OSS je programový produkt oddelený od licenciovaného, riadi sa vlastnými pravidlami pre voľne šíriteľný softvér (Open Source Definition), najznámejšie štandardy sú GNU General Public License (GPL) a Berkeley Software Distribution (BSD). Vyzdvihnem jeho výhody oproti Closed Source Software (CSS) – voľný preklad komerčný softvér – a pravdaže neobídem ani jeho nevýhody v druhej časti eseje.

Rozmer zdrojového kódu poskytuje užitočnú informáciu pre stanovenie jeho kvality, pričom hlavnými charakteristikami kvality sú napríklad udržovateľnosť, spoľahlivosť, rozšíriteľnosť a prenosnosť. Medzi hlavné požiadavky používateľov softvéru patrí jeho kvalita a už spomenuté charakteristiky kvality, preto sa touto témou bude zaoberať posledná časť.

## Úvod

Ak sa o nejakom programe povie, že je "open source" alebo "s otvoreným kódom", znamená to, že jeho zdrojové kódy sú prístupné každému. Nemusí to automaticky znamenať, že program je zadarmo, ani to, že možno tento kód ľubovoľne použiť.

V prvom rade sa pokúsím ujasniť, čo to vlastne je ten Open Source Software (OSS). Pre úplnosť najskôr popíšem všetky typy softvéru, ktoré sa vyskytujú na "trhu" so softvérovými produktami.

*Proprietárny softvér* je softvér, ktorého licencie nezaručujú užívateľom všetky práva definované pri open source software.

*Komerčný softvér* je softvér, ktorý je vyrábaný za účelom zisku. Komerčné produkty môžu byť zároveň otvoreným alebo slobodným softvérom.

*Freeware softvér* je softvér, ktorý je možné neobmedzene používať, avšak užívateľ nemá právo vidieť zdrojové kódy a právo softvér modifikovať.

*Manažment v softvérovom inžinierstve, máj 2005, s. 1-9.*

## Čo je to open source?

Open source softvér: voľne sa dá tento názov preložiť ako voľne šíriteľný, otvorený softvér, ktorého licencie zaručujú užívateľom základné práva, ako sú

- Dostupnosť zdrojového kódu: program musí obsahovať zdrojový kód a musí umožňovať distribúciu vo forme zdrojového kódu. V opačnom prípade musí byť definované, kde možno zdrojový kód programu získať bez poplatkov.
- Voľná redistribúcia: softvér musí byť voľne distribuovaný a za túto distribúciu nie je možné požadovať poplatky (výrazne) prevyšujúce náklady na túto distribúciu. Distribúciu nie je možné viazať na iné produkty.
- Voľná modifikácia: licencia musí umožňovať modifikovať softvér a odvádzať z neho nové produkty.
- Neobmedzené použitie: použitie softvéru nemôže byť viazané na určitú skupinu ľudí, komerčné, či nekomerčné použitie a podobne. Takisto nemôže byť obmedzené na použitie softvéru pri určitých činnostiach (genetický alebo jadrový výskum a podobne).

OSS vznikol len nedávno a za krátky čas bol implementovaný do mnohých produktov. Medzi najznámejšie patria operačný systém Linux, webservice Apache, internetový prehliadač Mozilla a ďalšie známe produkty s otvoreným kódom. Mnohí však stále netušia, k čomu "otvorený kód" slúži a čím sa odlišuje od ostatných softvérových produktov, skupinu tých ktorí to dobre vedia a práve preto na tento typ softvéru nedajú dopustiť tvoria jeho vývojári.

## Vývojári

Vývojári OSS produktov sú s týmito projektmi bytostne spojení, preto sa pokúsím opísať kto sú to, čo ich vedie k tomu aby investovali svoj voľný čas v prospech ostatných. Jednoznačne a nie prekvapivo prevládajú muži. Ich podiel voči nežnému pohlaviu je 98,9%. Ďalším zaujímavým údajom je vek, ktorý sa pohybuje medzi 14 až 73 rokmi. Pričom aritmetický priemer je na úrovni 27,1, čo jasne naznačuje, že prevažná väčšina ľudí aktívnych v open source je značne mladá.

Vzdelanie	% podiel
Univerzitné	70
Stredoškolské	14
Základné	8
Iné	8

**Tab. 1.** Úroveň vzdelania vývojárov

Tabuľka č. 1 ukazuje percentuálny podiel stupňa vzdelania v tejto skupine, avšak vzhľadom na celkový nízky vek vývojárov je veľká pravdepodobnosť, že absolventi základnej alebo strednej školy budú vo svojom vzdelaní pokračovať ďalej. Neprekvapujúco je väčšina vývojárov zamestnaná v IT sektore, prípadne na univerzitách. Najväčšiu skupinu s tretinovým podielom tvoria softvéroví inžinieri, nasledovaní 16% študentov. Zaujímavý je aj pohľad na nástroje, ktoré vývojári open source používajú. Vedúcou platformou je Linux na čele s konzervatívnou distribúciou Debian (41,1%). Nasleduje s veľkým odstupom RedHat, Mandrake a SuSE. Platformu Windows neprekvapivo používa len 2,2% vývojárov. Ešte spomeniem, že medzi najobľúbenejší textový editor v tejto skupine vývojárov patrí vi. Na záver tejto časti, v ktorej som sa venovala vyjadreniu kto sú to vlastne tí, ktorí stoja za OSS projektmi, musím spomenúť to najdôležitejšie, a to čo ich k tomu vedie. Táto otázka je veľmi zaujímavá, napríklad len 1,9% to nevie, až 78,9% vývojárov sa v open source učí. Táto skutočnosť odpovedá aj na otázku, čo dáva open source vývojárom. Azda každý mi dá za pravdu, že najlepšie sa učí na príkladoch, prípadne modifikáciou už existujúceho kódu. Ďalším z majoritných dôvodov je potreba podeliť sa o svoje vedomosti a skúsenosti, nasleduje chuť participácie inou, povedzme alternatívnou formou.

## Definovanie Open source

Keďže OSS je programový produkt oddelený od licenciovaného, riadi sa vlastnými pravidlami pre voľne šíriteľný softvér (Open Source Definition). Termín "open source" sa najčastejšie spája s "Open Source Initiative" (OSI). Je to nezisková organizácia, ktorá sa venuje propagovaniu otvoreného kódu, hlavne prostredníctvom certifikačného programu. V ňom posudzuje, či licencia programu spĺňa definíciu otvoreného kódu. Tá je určená splnením deviatich podmienok.

Pravdepodobne najznámejším licenčným štandardom pre otvorený kód je GNU General Public License (GPL) a Berkeley Software Distribution (BSD).

Projekt GNU (ktorý znamená GNU's Not UNIX -- GNU nie je UNIX, čo je rekurzívna skratka) založil pán Stallman, ktorý bol pôvodne pracovníkom MIT. Začal predávať univerzitám pásky so svojim softvérom -- čiže začal na ňom zarábať. Vymyslel licenciu GNU/GPL, ktorá zabezpečuje užívateľovi softvéru práva na jeho modifikovanie a podobne, zároveň však zaručuje, že nikto takýto softvér nevezme a nezačne ho s pár proprietárnymi zmenami predávať bez práv, ktoré s ním prišli. Neskôr pán Stallman založil Free Software Foundation (Nadáciu slobodného softvéru). Keď mal už skoro hotový celý systém GNU (chýbalo už len jadro), prišiel fínsky študent Linus Torvalds a napísal Linux. V tento deň vzniklo niečo, čo by sa po správnosti malo nazývať GNU/Linux (systém GNU s jadrom Linuxu).

Slobodný (alebo tiež voľný) softvér má podľa definície z Nadácie slobodného softvéru zaručovať tieto práva:

- Právo nula je právo spúšťať program za akýmkoľvek účelom, akýmkoľvek spôsobom chcete.

- Právo jedna je právo pomôcť si sám zmenou programu, aby vám čo najlepšie vyhovoval.
- Právo dva je právo pomôcť svojmu priateľovi distribuovaním kópií tohto programu.
- Právo tri je právo pomôcť vybudovať komunitu publikovaním zdokonalenej verzie, takže ostatní môžu využiť vašu prácu. Ak máte všetky tieto slobody, program je pre vás slobodný softvér.

Pán Ballmer z Microsoftu nazval softvér pod touto licenciou „rakovinou, čo sa nalepí na všetko, čoho sa dotkne“. Vlastnosťou tejto licencie je totiž, že akýkoľvek program odvodený od GNU/GPL je automaticky pod touto licenciou. Čiže v podstate žiadna firma nemôže legálne zobrať kód programu, pridať k nemu pár inovácií a tým pádom vziať ostatným, už spomenuté štyri základne práva. GNU/GPL však nezabraňuje predávaniu softvéru a rovnako neinfikuje ani výstup, ktorý za pomoci týchto programov dostaneme (napr. ak skompilujeme program kompilátorom gcc, ktorý je pod GNU/GPL, licenciu výstupného súboru to nijako neovplyvní). GNU/GPL teda zabezpečuje, že ak dostávame základné slobody, využívame ich, nemôžeme pri odvodennej práci (čiže modifikovanom programe) tieto slobody nikomu inému odoprieť. Je tu ešte jedna veľmi podstatná vec: ak daný kód zmeníme vo svoj prospech nemusíme ho šíriť ďalej, môžeme si ho ponechať v našom vlastníctve a nikomu ho neposkytnúť. To je naše právo. Ak ho šírimo, šírimo ho so všetkými slobodami.

Druhou štandardou je BSD licencia. Takto sú licencované veľké časti systémov FreeBSD, NetBSD a OpenBSD. Prakticky hovoria, že si môžeme so softvérom robiť všetko, ak necháme v zdrojovom texte pôvodnú poznámku o copyrighte a samozrejme rovnako ako GNU/GPL pozbavujú autora kódu akejkolvek zodpovednosti za prípadné škody. Veľa ľudí oponuje tejto časti licencií slobodného softvéru s argumentom, že takéto niečo si napr. banka nemôže nikdy dovoliť nasadiť. Treba poznamenať, že väčšina proprietárneho softvéru má licenciu, ktorá autorovi ukladá zodpovednosť len do výšky ceny softvéru. Takže ak používame webserver za dvesto tisíc korún a niekto nám cez tento softvér vykradne bankové konto a my budeme mať škodu napr. dve miliardy korún, autor nám vráti len, tých dvesto tisíc korún.

A tu sa dostávam k Ericovi Raymondovi a jeho dielu *The Cathedral and the Bazaar*. Pán Raymond opísal vývojový model okolo slobodného softvéru a s niekoľkými ďalšími ľuďmi založili, už spomínanú Open Source Initiative. Mysleli si, že anglický termín Free Software (čo môže znamenať slobodný softvér, ale väčšina ľudí si to prekladá ako softvér, ktorý je zadarmo) je príliš mäťúci. Títo páni sa rozhodli, že budú certifikovať softvérové licencie na základe podobných pravidiel ako Free Software Foundation, ale budú mu dávať nálepku Open Source softvér. Rozdiel medzi open-sourcistami a slobodným softvérom je pomerne malý, ale aj tak: open-source hnutie sa zameriava len na jednu výhodu slobodného softvéru: na lepší vývojový model, ktorý produkuje kvalitnejší kód ako proprietárne programy. Definícia open source softvéru je tiež pomerne jemnejšia a spadajú pod ňu aj niektoré licencie, ktoré FSF neklasifikuje ako slobodný softvér. Tam však rozdiely končia a oba „tábory“ v podstate na vývoji softvéru spolupracujú.

Ako sa možno dočítať na stránke OSI: "Základná myšlienka otvoreného kódu je veľmi jednoduchá: keď programátori môžu čítať, redistribuovať a modifikovať zdrojový kód nejakej časti softvéru, tento softvér sa vyvíja. Ľudia ho vylepšujú, prispôbujú a opravujú chyby. Toto všetko sa deje rýchlosťou, ktorá je pre človeka zvyknutého na pomalý vývoj konvenčného softvéru udivujúca."

Pre používateľov programov s otvoreným kódom z toho vyplýva vyššia kvalita, rýchlejší vývoj a hlavne oveľa rýchlejšie opravovanie chýb, než je tomu u softvéru s uzavretým kódom. Oprava pre akúkoľvek chybu je hotová najneskôr o niekoľko hodín, občas je dokonca súčasťou správy o chybe. Keďže kód je verejne prístupný, môže opravu urobiť ktokoľvek. Ak sa chyba objaví v uzavretom kóde, treba počkať na autora, kým ju opraví on sám, čo môže trvať dosť dlho.

Otvorený kód programu nemá pre absolútnu väčšinou používateľov žiadne nevýhody. Iba vo výnimočných prípadoch môže jeho otvorenosť prekážať, hlavne ak sa vyžaduje paranoidná úroveň bezpečnosti (firmy a organizácie, ktoré ju vyžadujú, si však väčšinou práve z tohto dôvodu vytvárajú softvér vo vlastnej réžii).

Prečo je potom softvéru s otvoreným kódom tak málo? Lebo nie je výhodný pre softvérové firmy. Ak je pre nich softvér hlavným zdrojom príjmov, nemôžu zverejniť jeho zdrojový kód. Hocikto by ho potom mohol skopírovať a firma by prišla o príjmy. V prípade softvéru s veľkým množstvom potenciálnych používateľov (operačné systémy, office aplikácie a podobne) sa veľkým firmám oplatí sponzorovať ich tvorbu aj s otvoreným kódom. V oblasti vysoko špecializovaného a na konkrétne požiadavky prispôbeného softvéru otvorený kód nájdete iba výnimočne.

## Rozmery zdrojového kódu

Dobre známym predpokladom v súčasnom softvérovom inžinierstve je, že externé znaky kvality sú súvzťažné k interným (vnútorným) znakom. Veľkosť zdrojového kódu poskytuje užitočnú informáciu pre odhad jeho vlastností, pre odhad vonkajších systémových znakov kvality, ako sú udržovateľnosť, spoľahlivosť, rozširiteľnosť a prenositeľnosť.

Tieto rozmery môžu byť použité pre získanie obrazu o kvalite, tak jednotlivých komponentov, ako aj celého programu. Typickou softvérovou metrikou je veľkosť kódu (vymeraný v riadkoch kódu, v množstve dedení, a tak ďalej) a kódová zložitosť.

Softvérové organizácie zainteresované do vývoja programového vybavenia zostavili vlastné štandardy. Príkladom je štandard NSA [3], ktorý je odvodený z analýzy 25 miliónov riadkov programu napísaného pre NSA.

## Meranie a oceňovanie Open source kódu

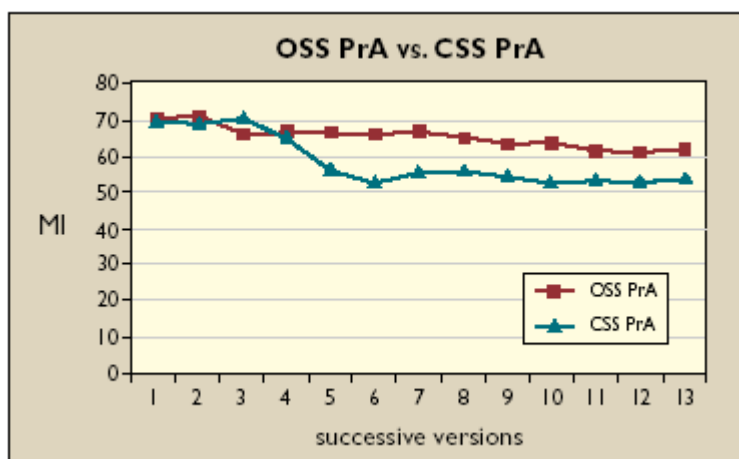
Nasleduje porovnanie piatich aktívnych a obľúbených OSS projektov zo zaujímavým životným cyklom. Tab. 1 poskytuje typické projektové informácie (aplikačný typ, celková kódová veľkosť) a krátky slovný popis projektového vývoja.

Projekt	Aplikačný typ	Veľkosť kódu (KLOC)	Popis projektu
OSSPrA	Operačný systém	343	OSS projekt, ktorý bol navrhnutý ako CSS, ale stále vyvíjaný ako OSS
CSSrA	Operačný systém	994	CSS projekt, ktorý bol navrhnutý ako OSS, ale vyvíjaný ako komerčný. Predstavuje protiklad k OSSPrA.
PrB	Operačný systém	860	CSS projekt, ktorý bol pretransformovaný na OSS projekt.
PrC	Programovací jazyk	1050	Čisto OSS projekt.
PrD	Databázový systém	1411	Akademický projekt, ktorý vznikol ako OSS projekt
PrE	Internetová aplikácia	1198	Čisto OSS projekt

**Tab. 2.** Päť open source projektov a ich pozoruhodné charakteristiky

Neudávam skutočné názvy projektov, kvôli dodržaniu štandardov etiky softvérového inžinierstva [5], preto je každý projekt označený len všeobecne.

Tu predstavujem výsledky, ktoré sa týkajú indexu udržateľnosti (Obr. 1, 2, a 3), ktorý umožňuje rýchlu analýzu projektov.



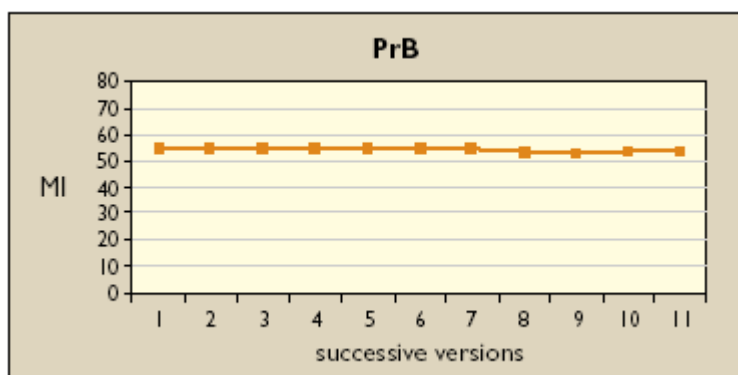
**Obr. 1.** Hodnoty MI pre projekt PrA.

Projekt PrA bol vytvorený ako originálny OSS projekt. Neskôr sa riadiaca skupina rozhodla využiť systém komerčne a spustila projekt typu close source (označeného ako CSSPrA).

Ale, originálny OSS systém (OSSPrA) neprestali vyvíjať. Funkčnosť oboch projektov v dobe tvorby tohto porovnávania bola zhruba rovnaká. Tento fakt vytvoril

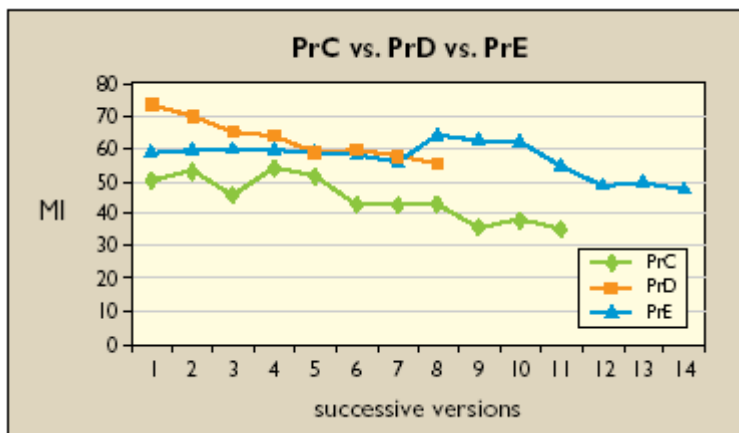
zaujímavú situáciu, ktorá dovolila vytvorenie priameho porovnania medzi OSS a CSS projektom, ktoré rozvíjali ten istý objekt. Obr. 1 poskytuje veľmi zaujímavé hodnoty: zdá sa, že udržovateľnosť sa zhorší v oboch prípadoch, avšak OSS verzia sa javí významne lepšie ako jej CSS náprotivok.

Projekt PrB bol originálne CSS projekt. Časom sa spoločnosť, už počas rozvíjania systému, rozhodla zverejniť zdrojový kód, čo viedlo k jeho prechodu na OSS projekt. Projekt PrB poskytuje situáciu v ktorej je CSS objekt vyvíjaný v OSS móde. Obr. 2 dáva predstavu o tom, čo sa stane s udržovateľnosťou systému v nasledujúcich etapách oproti počiatočnému vydaniu. Následkom je dosť rýchly ďalší vývoj projektu, udržovateľnosť nových verzií očividne nevykazuje významné rozdiely vzhľadom na prvé uverejnenie, ktoré bolo v skutočnosti vypracované ako CSS. Pre sledovanie stavu udržovateľnosti systému je potrebné aj naďalej sledovať vývoj tohto projektu, aby sa dosiahli správne a presné výsledky. Aj keď, predbežne možno povedať, že tento prípad ukázal ako prechod od CSS k OSS nezhorší hodnoty udržovateľnosti.



Obr. 2. Hodnoty MI pre projekt PrB

Projekty PrC a PrE sú typické OSS projekty: ktoré boli zahájené ako OSS projekty a neustále boli vyvíjane ako také. PrD bol pôvodne akademický projekt, ktorý sa vyvinul na OSS projekt (len OSS vydanie bolo zobrazené do úvahy). Obr. 3. poskytuje históriu udržovateľnosti týchto projektov v jednoduchom diagrame: udržovateľnosť klesne postupne vo všetkých troch prípadoch. Len v PrE sa podarilo zlepšiť udržovateľnosť pre limitovaný počet vydaní. Zdá sa, že tieto projekty sa chovajú ako CSS projekty.



Obr. 3. Hodnoty MI pre projekt PrC, PrD, PrE.

## Záver

Z hľadiska používateľa sa programy s otvoreným kódom prakticky nijako nelíšia od ostatných, aj pravdepodobnosť výskytu chýb v nich je v podstate rovnaká. Ich výhodou je, že väčšinu chýb sa podarí objaviť v krátkom čase a prakticky hneď sú aj odstránené. Vďaka tomu existuje niekoľko programov s otvoreným kódom, v ktorých nebola objavená žiadna závažnejšia chyba už niekoľko rokov.

Niekedy sa svet open source softvéru označuje termínom „from boom to dust“, čo voľne preložené znamená „z hviezd do prachu“. Avšak tento prívlastok je dosť prehnaný – OSS nikdy nebol „vo hviezdach“ a ani teraz rozhodne nie je „v prachu“. OSS je práve na „jedálnom lístku“ trhu informačných technológií a je posudzovaný podľa úplne rovnakého komplexu vlastností a parametrov ako akýkoľvek iný systém, či riešenie. OSS opustil ideológiu („free ako freedom“) a je súčasťou ekonomiky („free ako free beer“). Pokiaľ zástancovia free softvéru dnes vášnivo upozorňujú, že free neznamená „zdarma“, ale „slobodný“, sú dosť mimo, pretože ktokoľvek, kto bude vyhodnocovať nasadenie OSS, bude na prvom mieste vyhodnocovať to, že je „zdarma“ a technické parametre. To, že je tu možnosť si softvér upraviť bude pre väčšinu úplne v pozadí. OSS nesmierne ovplyvnil svet softvéru a jeho vplyv pokračuje.

## Použitá literatúra

1. Bednár Juraj: O slobodnom softvéri a open source. *PC Revue*, Vol. 8, No. 8 (2001) 22-23.



2. Drake, T. Measuring software quality: A case study. *IEEE Computer* 29, 11 (1996), 78–87.
3. El-Emam. Ethics and open source. *Empirical Software Engineering* 4, 6 (2001), 291–292.
4. Ioannis Somoladas, Ioannis Stamelos, Lefteris Angelis and Apostolos Oikonomou: Open Source Software Development Should Strive for even greater code maintainability. *In Communication of the ACM*. Vol. 47. No. 10 (October 2004). 83-87.
5. Willmann Richard: Kto je to Open Souce? *PC Revue*, Vol. 9, No. 10 (2002) 127-128.

### **Annotation**

#### *Open Source Software*

The main goal of this essay is approximate and explain to reader, what does mean Open Source Software, it's creation, introduction of its authors and explaining of reasons why people develop OSS.

OSS is separated from licensed product, so it uses own rules for open source software (Open Source Definition). The most famous standards are GNU General Public License (GPL) and Berkeley Software Distribution (BSD). In the this part is compared with CSS (commercial system) - its pros, in the second part its cons.

The size of source code gives useful information about its quality (the main characters of quality are maintainability, reliable, expandability and flexibility. The main user's requirements are quality and already named characteristics, that's why it is written in the last part of this essay.