

Debugovanje Java programa

Ivan Vranković, Anđela Karakaš, Ivan Mihajlović, Kristina Petrović

20. april 2020.

Sadržaj

1 Uputstva	2
2 Recenzent — ocena:	3
2.1 O čemu rad govori?	3
2.2 Krupne primedbe i sugestije	3
2.3 Sitne primedbe	3
2.4 Provera sadržajnosti i forme seminarskog rada	4
2.5 Ocenite sebe	5
3 Recenzent — ocena:	6
3.1 O čemu rad govori?	6
3.2 Krupne primedbe i sugestije	6
3.3 Sitne primedbe	7
3.4 Provera sadržajnosti i forme seminarskog rada	7
3.5 Ocenite sebe	9
4 Recenzent — ocena:	10
4.1 O čemu rad govori?	10
4.2 Krupne primedbe i sugestije	10
4.3 Sitne primedbe	12
4.4 Provera sadržajnosti i forme seminarskog rada	14
4.5 Ocenite sebe	15
5 Dodatne izmene	16

Glava 1

Uputstva

Prilikom predavanja odgovora na recenziju, obrišite ovo poglavlje.

Neophodno je odgovoriti na sve zamerke koje su navedene u okviru recenzija. Svaki odgovor pišete u okviru okruženja \odgovor, [kako bi vaši odgovori bili lakše uočljivi](#).

1. Odgovor treba da sadrži na koji način ste izmenili rad da bi adresirali problem koji je recenzent naveo. Na primer, to može biti neka dodata rečenica ili dodat pasus. Ukoliko je u pitanju kraći tekst onda ga možete navesti direktno u ovom dokumentu, ukoliko je u pitanju duži tekst, onda navedete samo na kojoj strani i gde tačno se taj novi tekst nalazi. Ukoliko je izmenjeno ime nekog poglavlja, navedite na koji način je izmenjeno, i slično, u zavisnosti od izmena koje ste napravili.
2. Ukoliko ništa niste izmenili povodom neke zamerke, detaljno obrazložite zašto zahtev recenzenta nije uvažen.
3. Ukoliko ste napravili i neke izmene koje recenzenti nisu tražili, njih navedite u poslednjem poglavlju tj u poglavlju Dodatne izmene.

Za svakog recenzenta dodajte ocenu od 1 do 5 koja označava koliko vam je recenzija bila korisna, odnosno koliko vam je pomogla da unapredite rad. Ocena 1 označava da vam recenzija nije bila korisna, ocena 5 označava da vam je recenzija bila veoma korisna.

NAPOMENA: Recenzije ce biti ocenjene nezavisno od vaših ocena. Na osnovu recenzije ja znam da li je ona korisna ili ne, pa na taj način vama idu negativni poeni ukoliko kažete da je korisno nešto što nije korisno. Vašim kolegama šteti da kažete da im je recenzija korisna jer će misliti da su je dobro uradili, iako to zapravo nisu. Isto važi i na drugu stranu, tj nemojte reći da nije korisno ono što jeste korisno. Prema tome, trudite se da budete objektivni.

Glava 2

Recenzent — ocena:

2.1 O čemu rad govori?

U ovom radu je obrađena tema debugovanja u Javi. Rad polazi od osnovnih koncepata debugovanja i različitih pristupa debugovanju, nakon čega поближе opisuje funkcionisanje različitih tipova debagera i njihovih glavnih predstavnika. Kao primer debugovanja u Javi, ovaj rad se usmerio ka debugovanju serverskih aplikacija i prikazuje neke od čestih propusta koji se javljaju pri programiranju u Javi.

2.2 Krupne primedbe i sugestije

Radu svakako nedostaje uvod i zaključak, kako bi se stekao bolji utisak koji su ciljevi ovog seminarskog rada, kao i kakva su konačna zapažanja. Ono što zapažam je čudna raspodela odeljaka. Naime, čitalac se odmah u prvom poglavlju uvodi u koncepte deklarativnog debugovanja, da bi se tek u trećem poglavlju detaljno posvetili tipovima grešaka u Java programima.

2.3 Sitne primedbe

Rad je nekonzistentno pisan čas ošišanom latinicom, čas punom latinicom, a i slovo “đ” je na dosta mesta “progutano”. Autor ove recenzije je primetio i slovne greške u radu (npr. piše se “komponenata”, a ne “konponenata”). Mešanje anglicizama i sprskih prevoda računarskih termina je takođe učestalo (npr. u istom pasusu se koriste reči “kompajliranje” i “prevodilac”). Kodovi nisu ni na koji način izdvojeni iz teksta, što doprinosi nečitkosti rada. Takođe, tekst nije centriran u odnosu na stranice, a u određenim linijama je došlo do “probijanja” teksta van predviđenih granica.

2.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
Da, rad upoznaje čitaoca sa svim načinima na koji Java program može da se debuguje.
2. Da li je nešto važno propušteno?
Ne, rad obuhvata osnovne koncepte debugovanja Java programa.
3. Da li ima suštinskih grešaka i propusta?
Ne, rad odgovara na temu debugovanja Java programa osvrćući se na različite pristupe pri debugovanju Java programa.
4. Da li je naslov rada dobro izabran?
Da, naslov rada odgovara sadržaju seminarskog rada.
5. Da li sažetak sadrži prave podatke o radu?
Da, sažetak je obuhvatio sve što se nalazi u seminarskom radu.
6. Da li je rad lak-težak za čitanje?
Što se tiče tematike koju rad obrađuje, smatram da nije težak za čitanje. S druge strane, čudno formulisane rečenice jesu pravile problem prilikom razumevanja rada.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Za razumevanje teksta je potrebno osrednje poznavanje programskog jezika Java, poznavanje mehanizama Java programskog jezika, kao i osnovno poznavanje principa debugovanja.
8. Da li je u radu navedena odgovarajuća literatura?
U radu je navedena literatura, ali u istoj se nalaze i linkovi ka forumima kao što je StackOverflow.
9. Da li su u radu reference korektno navedene?
Ne, rad nije referenciran.
10. Da li je struktura rada adekvatna?
Ne, rad nema uvod i zaključak. Takođe su i poglavlja čudno formatirana (npr. drugo poglavlje sadrži svega 6 redova)
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
Rad sadrži tabelu, a broj strana je u skladu sa propisanim uslovima. Međutim, slika je uvezena u rad, ali nije vidljiva u pdf-u koji je dat za recenziranje. Postoji i dijagram toka, ali nije referisan nigde u radu.
12. Da li su slike i tabele funkcionalne i adekvatne?
Tabela je smisljena i čitljiva. Dijagram toka je u određenoj meri razumljiv, ali ne na prvi pogled.

2.5 Ocenite sebe

Što se tiče oblasti koju recenziram, smatram da sam skoro neupućen po pitanju debugovanja u Javi. Moje iskustvo (koje sam na fakultetu do sada stekao na dva kursa) se zasniva na najprostijem obliku debugovanja u Javi, tj. nisam se bavio debugovanjem koje zalazi “ispod haube”.

Glava 3

Recenzent — ocena:

3.1 O čemu rad govori?

Rad govori o debugovanju i problemima sa kojima se programeri susreću prilikom rada sa: Java programskim jezikom, Java virtuelnom mašinom (JVM), i Java ekosistemom za serverske aplikacije. Najveći deo rada posvećen je tipovima debagera u Javi.

3.2 Krupne primedbe i sugestije

Radu nedostaju uvod i zaključak. Prvi deo rada koji bi trebalo da predstavlja uopšteni uvod na temu “Šta je debugovanje?” takođe nedostaje i rad odmah počinje sa debugovanjem u Javi.

Na nekoliko strana tekst izbija van margine, a cela desna margina je prevelika. Na strani 9 i 10, primeri koda nisu formatirani da izgledaju kao java kod. Slika na strani 10 nije dovoljno opisana i nije očigledno koja je poenta i šta bi trebalo da predstavlja. U radu nema ni jedne reference niti bibliografije. Izvori su samo nalepljeni na kraj rada.

U poglavlju “1 Java debugovanje” poslednja rečnica: “Radi dodatne sigurnosti jezik je proširen elementima kao što su: sakupljač otpadaka (eng. garbage collection), provere granica nizova, interfejsi i paketi.” je formulisana tako da tvrdi da Java nije imala ove funkcionalnosti od samog početka, a jeste.

U podpoglavljju “1.2 Debugovanje konkurentnih Java programa” pasus koji govori o metodi `synchronized()` nema veze sa debugovanjem konkurentnih programa već sinhronizacijom. Ovo poglavlje bi trebalo da se bavi debugovanjem nastalih grešaka, a ne izbegavanjem jer to spada u oblast konkretnog programiranja.

Tekst podpoglavlja “1.3 Testiranje na različitim platformama” se ne slaže sa naslovom. Tekst govori o kačenju debagera na aktivan proces i udaljenom debugovanju umesto o problemima koji mogu nastati pri pokretanju Java programa na različitim platformama i kako se testira da se otkriju ako ih ima.

Poglavlje “2 Java debager” nema nikakvu suštinu. Piše da se će u ovom odeljku biti prikazane razlike među vrstama debagera, a nema prikaza. Rečnica da izbor debagera zavisi od situacije nije nešto što treba da se napiše jer se podrazumeva, informativno bi bilo napisati od kojih situacija zavisi i koji bi debager trebalo odabrati u zavisnosti od neke specifične situacije.

U poglavlju “3 Kategorije i tipovi grešaka“ je utisak prilikom čitanja kao da se svaki paragraf prerano završio. Možda bi trebalo dodati primere grešaka koje najčešće prave Java programeri početnici i kako se one mogu debugovati.

U podpoglavljju 4.1.3 rečenica “Može se pokrenuti na serveru veoma brzo bez potrebe za skladištenjem samog debagera na serveru“ zvuči kontradiktorno. Ako se nešto pokreće na serveru onda to implicira da se tamo i nalazi. Možda treba preformulisati da bolje opiše debugovanje na daljinu. Tvđenje da su alati komande linije i jdb među njima previše kompleksni za svakodnevno korišćenje je subjektivna. Većina alata komandne linije postoji i koristi se baš zato što su jednostavniji nego grafički korisnički interfejs.

U poglavlju “4.2 Debageri integrisani u razvojno okruženje“ se spominju u najboljem slučaju drugorazredna integrisana okruženja za razvoj: Eclipse i NetBeans, a ne spominje se JetBrains-ov IntelliJ koji je trenutno najnaprednije razvojno okruženje za Java ekosistem, daleko ispred Eclipse i NetBeans-a. Nedostatak koji je naveden za debugovanje na daljinu nije tačan za IntelliJ. Debugovanje na daljinu u IntelliJ se ne razlikuje značajno od standardnog debugovanja.

Nakon poglavlja “5 Debugovanje serverskih aplikacija“ se odmah nalazi pod poglavljje “5.1 Debugovanje servleta“. Sadržaj ispod naslova poglavlja ne bi trebalo da bude prazan i da odmah posle njega sledi naslov podpoglavljja. Isti komentar i za poglavljje 6.

Debugovanje ne obuhvata proces ispravljanja grešaka prevodioca. To nisu bagovi (neželjna ponašanja). Upravo suprotno, greške koje prevodilac izbaci su poželjne.

Poglavljju 6.2 o fali više sadržaja i nema prikaza primene teorijskog znanja na konkretnom primeru kao što se to navodi u sažetku rada.

3.3 Sitne primedbe

Mešaju se stilovi pisanja. Druga rečenica sažetka “U nastavku cemo se baviti detekcijom bagova[...]“ je u prvom licu množine, a neka od narednih je u pasivu. Sve rečenice bi trebalo da budu u pasivu. Umest “U nastavku ćemo se baviti detekcijom bagova“ pasiv bi bio “Rad obrađuje detekciju bagova, tehniko otklanjanja ...“. Kroz celokupan rad su rečenice u različitim licima.

Latinična slova sa kvačicama: č, ć, đ, i ž; su na više mesta napisana kao c, dj, i z. Na više mesta nema razmaka nakon tačke, zareza, i stavke nabrajanja. Mnoštvo grešaka u kucanju kao na primer “konponenata“ umesto “komponenta“. U podpoglavljju 4. postoje rečenice koje počinju malim slovom. Korišćenje kvalifikacije bez konkretnog značenja: “slabo“, “jako“. Sažetak i sadržaj su ostali engleski nazivi.

3.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?

Ne stiče se utisak da rad odgovara na temu. Nema ničeg opštog o tome šta je debugovanje. Postoji mnoštvo ezoteričnih bagova sa kojima se Java programeri susreću (curenje memorije, bagovi u JVM), a nisu adekvatno

obrađeni. Ne spominje se IntelliJ koji je jedan od glavnih alata za debugovanje Java programa.

2. Da li je nešto važno propušteno?
Nedostaju uvod i zaključak i uvodni tekstovi u poglavljima 5 i 6.
3. Da li ima suštinskih grešaka i propusta?
Suštinski propust je to što nakon čitanja rada ne postoji zaključak koji se dobije iz rada. Rad bi trebalo da nudi neku korisnu informaciju koja čitaocu može da pomogne u donošenju neke odluke kada debuguje Java program.
4. Da li je naslov rada dobro izabran?
Naslov rada je dobro izabran. Poklapa se sa sadržajem. Tekst svakog poglavlja se ne poklapa uvek sa naslovom poglavlja.
5. Da li sažetak sadrži prave podatke o radu?
Sažetak daje prave podatke o radu. Ono što piše u sažetku može se videti i u sadržaju, bez čitanja sažetka.
6. Da li je rad lak-težak za čitanje?
Rad je težak za čitanje. Paragrafi nisu uvek pravilno razdvojeni i preklapaju se teme o kojima se priča u jednom paragrafu iz rečenice u rečenicu. U poglavljima 5 i 6 nema uvodnog teksta o čemu se radi u tom poglavlju već odmah počinj sa podpoglavljima.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Za razumevanje teksta je potrebno da je čitalac upoznat sa Java programskim jezikom, debugovanjem, radom u spomenutim tehnologijama iz Java ekosistema kao što su na primer servleti, i Java virtuelnom mašinom.
8. Da li je u radu navedena odgovarajuća literatura?
U radu od 10 stavki literature 5 se mogu smatrati kredibilnim i vrednim citiranja. To su:
 - (a) Java Servlet Programming, Beginning JavaServer Pages
 - (b) The Pragmatics Of Java Debugging
 - (c) <https://docs.oracle.com/>
 - (d) <https://www6.software.ibm.com/developerworks/education/j/j.pdf>Ostatak linkova sa interneta ne mogu se smatrati kredibilnim niti relevantnim za datu temu.
9. Da li su u radu reference korektno navedene?
Bibliografija ne postoji. Ne postoji ni jedna referenca u radu.
10. Da li je struktura rada adekvatna?
Fale uvod i zaključak. Ostali su engleski nazivi za sažetak i sadržaj. Bibliografija takođe fali. Ima poglavlja i podpoglavlja. Ima mesta gde pasus sadrži samo jednu rečenicu ili samo nabranje.

11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
Jedna slika nije učitana. Postoji jedna tabela, ali bez labele. Broj strana je ispunjen.
12. Da li su slike i tabele funkcionalne i adekvatne?
Tabelama i slikama fale opisi (eng. caption). Nedostaje slika "Figure 1. JDPa" na strani 6. Tabela na strani 4 daje pregled osnovnih komandi i šta one rade. Nigde nije referencirana u tekstu niti ima opis.
Poenta slike na strani 10 nije očigledna. Niti postoji objašnjenje. Konkretno bi trebalo objasniti šta znači korak dijagnostikovanja. Raspored figura na toj slici ne daje jasan tok procesa debugovanja. Jasnije bi bilo kada bi strelice i figure bile raspoređene tako da tok ide odozgo na dole.

3.5 Ocenite sebe

Rekao bih da sam srednje upućen u oblast. Napisao sam seminarski rad na temu poređenja razlika između Java virtuelne mašine i C# virtuelne mašine tako da znam ponešto o funkcionisanju okruženja na kojima se izvršavaju ovi jezici i kakvih bagova ima u tim okruženjima. Susreo sam se sa JDK-4726957 bagom u Java virtuelnoj mašini koji dovodi do neuspeha prilikom zatvaranja Socket-a metodom close.

Glava 4

Recenzent — ocena:

4.1 O čemu rad govori?

Rad nastaje kao odgovor na neizbežnu pojavu raznoraznih bagova tokom pisanja u Javi. Od jednostavnih i kratkih kodova do složenih sistema, čitanjem uvidamo šta sve može poći po zlu, kako postojeći bag detektovati i zavisno od naših potreba koju metodu koristiti. Posebno je korisno pročitati deo vezan za serversko debugovanje, koje je drugačiji od dosad viđenih stvari i specifičan za Javu.

Shodno neophodnosti i koristi korišćenja CLI metode tokom produkcionog procesa i kod problema na serveru, posvećen je veći deo ovom pristupu u debugovanju, uz sve osnovne komande koje nam mogu zatrebati i metode koje se mogu primenjivati. Zatim su kao dopuna CLI metodi opisani i debageri integrisani u razvojna okruženja, gde se NetBeans i Eclipse ističu kao najpopularniji i debageri istorije koji su jako korisni za testiranje, pogotovo onda kad nam je potrebno da imamo dokumentovana stanja izvršavanja i želimo preko njih da reprodukujemo greške. Veliku slobodu i kontrolu tokom testiranja koda koji je u fazi izvršavanja može nam dati dinamičko praćenje, a ako pravimo veliki sistem, mogu nam poslužiti i frejmvorci za logovanje i log analizatori. Ako pak i to nije dovoljno, data je kratka uvodna priča o JDBA(Java Platform Debugger Architecture) kojim se mogu napraviti sopstveni debageri.

Zbog svoje složenosti i višenitnosti, posebno poglavlje posvećeno je debugovanju servera i pokretaču servleta koji kompanija Sun obezbeđuje, zajedno sa osnovnim načinima korišćenja istog, kao i debugovanju serverskih stranica i problema koji mogu nastati njihovim prebacanjem u servlete i debugovanjem.

Na samom kraju demonstrirani su neki kratki primeri koji dovode do sintaksnih grešaka i izuzetaka i pojašnjeno je kako čitati i iskoristiti tekst tih grešaka koji nam izlazi u terminalu. Što se tiče pomenutih složenijih metoda, oni nisu dočarani primerima, već je preneto samo njihova suština, dovoljna za ideju o njima i inicijalno upoznavanje.

4.2 Krupne primedbe i sugestije

Ono što mi se dopalo je to što se autori nisu mnogo zadržali na uvodnom delu i opštoj priči o debugovanju. Posle kratkog uvoda o bagovima od par rečenica,

u radu se odmah prelazi na debugovanje u Javi. Navede se da od situacije zavisi koji ćemo debager koristiti i to je dobar uvod za priču o njima. Međutim, taj tok biva prekinut i oni dolaze na red tek posle neugodno smeštene druge glave, posvećene definiciji bagova i podeli posledica bagova. Možda je prikladnije da se ova priča nalazi u uvodu? Tako bi rad imao prirodni tok i bilo bi manje ponavljanja.

Čitava četvrta glava jako striktno prati sam izvor i možda bi bilo bolje reorganizovati je i ukombinovati sa nekim drugim izvorima, malo promeniti format: šta je, gde se koristi, prednosti, mane, u recimo format gde će se prednosti i mane istaći na kraju ili format gde se sve ovo neosetno provlači kroz tekst. Čisto da bude različito od izvora i čitljivije. Neke rečenice su malo negrapno prevedene sa engleskog, što je osetno tokom čitanja i valjalo bi to promeniti. Prekontrolisati i da li je sve dobro prevedeno. Recimo:

This means that historical debuggers can slow down an application anywhere from 50% to an order of magnitude

prevedeno je kao

što znači da usporeenje programa korišćenjem ovih debagera može biti od marginalnih, pa čak do 50% što je razlog....

Dakle, ovde kaže od 50, ne do.

Lepo što su stavljani i graf i obična fotografija, ali graf deluje nepotrebno veliki. Sigurna sam da se može smanjiti tako da bude u širini teksta, kao i da se srede obeležja u njemu, ispod referiše na njega i poveže tako sa tekstom koji bi trebalo da ga objasni. Tekst ispod je nezavisan od grafa, pa deluje da slika nije sa tim objašnjenjem povezana i prekida se tok u čitanju i zbunjuje čitalac. Možda bi bilo lepše i da je uzet ovaj graf iz literature, iako je na engleskom, ali da je objašnjen.

Nije ostvaren takav tok ni sa slikom, ne referiše se na nju i time povezuje slika sa komponentama, sa prethodnim tekstom gde su te komponente nabrojane. Ceo tekst posle slike koji treba sa njom da bude povezan je bukvalno i jako nezgrapno preveden sa engleskog. Možda pogledati istu stvar iz još nekog izvora, pa jasnije predočiti svojim rečima.

Tabela takođe nije referisana nigde u tekstu i malo je šira od širine teksta, čime se vizuelno kvari efekat. Tabela ima i horizontalne i vertikalne linije, koje ne izgledaju strašno, ali bi možda bilo dobro pokušati sa većim proredom između, ako se ne doda više teksta kasnije i ostane prostora.

Što se tiče dodavanja, u trećoj glavi bi bilo zgodno navesti sve funkcije analizatora hipa, jer su i one bitne u datom kontekstu, a ne samo jednu navedenu. U šestoj glavi, spominje se da ukoliko program u izvršavanju dođe do run-time grešaka, program se prekida i generiše se dijagnostika. Možda bi bilo dobro pomenuti da ove delove treba staviti u try-catch blokove da bi mogla da se ispiše dijagnostika. U većini literature se razlikuju greške i izuzeci. U ovom delu, prolaze se primeri za kompajlerske i run-time greške. Možda bi se mogao dodati i deo kad program radi, ali se izlaz razlikuje od očekivanog, tj. pokriti i logičke greske i *simbolički debager*. Bilo bi dobro imati i neke lične primere u radu.

Korišćeno je dosta skraćena za pomenute tehnologije, i za većinu njih navedeno je puno ime na engleskom. Međutim, možda bi bilo korisno dodati isto

i za skraćenice kao što je neobjašnjena QA(4.4) ili neki izvor za ad-hoc koji se pominje na nekoliko mesta, da bi sve bilo razumljivo i neiskusnom čitaocu.

Sam rad je celina za sebe i dobro predstavlja osnove za datu temu, ali traženje dodatnih informacija, kao i meni sad provera istih, otežano je zbog nereferisanja na literaturu. Literatura je samo pobrojana dole, u formi knjiga i linkova, bez ikakve reference u bilo kom delu samog teksta. Deluje mi da bibtex nije uopšte korišćen. Takođe, u sadržaju nisu linkovi, što otežava navigaciju. To bi bilo dobro dodati. Možda grešim, ali deluje mi da neki izvori nisu ni iskorišćeni(stack-how to handle runtime errors, u tom delu nema ničega odavde).

Za deliće programskog koda, kao i ispile u terminalu, možda bi bilo dobro koristiti neku Latex naredbu, a ne samo boldovanje, radi veće čitljivosti i elegancije.

4.3 Sitne primedbe

Čim uzmemo rad, na prvoj strani mogu se primetiti izmešan engleski i srpski, naime umesto abstrakt i sadržaj, stoje abstract i contents. Zatim, nezanemarljiv deo rada napisan je na ošišanom srpskom jeziku. Naime, umesto š, đ, ž, č i é korišćeni su s, d, z i c. U nekim delovima uporno je izostavljano slovo đ(meu, meutim, utvrjuju, prosleivanja, prosleenim, odreenom, takoe, naie, odrene, izmeu), dok je u drugim uporno pisano dj. Ima izostavljanja po jednog slova ili korišćenja pogrešnog(pokreće, imzedju, petantan, infomarcije, konponenata), nedostatak delova reči(linij, probl, teks, različitm, teća), pogrešno stavljenih belina(bi smo, okom, komje) ili kombinacije(ne navoenja).

Što se tiče vizuelnog efekta i čitljivosti, osim pomenutih problema sa slikom i grafovima, pozicija teksta kviri utisak, jer u celom radu, tekst nekako stoji više levo. Kao da nije centriran. Takođe, određene reči zbog svoje dužine izlaze iz granica teksta, na primer

- HeapDumpOnOutOfMemoryError(4.4)
- sun.servlet.http.HTTPServer()(5)
- primeri grešaka prevodioca (6)

Vizuelni efekat takođe kvare i ugnježdene zagrade koje se javljaju na nekoliko mesta u tekstu, kao i počinjanje rečenica malim jhat i jmap. To je nešto što je moguće zaobići uz reformulaciju teksta. Bilo bi dobro staviti i tekst između naslova i podnaslova(6), po savetima sa časa.

Pomenula bih da je otežana čitljivost u nekim delovima, recimo u 4.1 i 4.1.1, gde se posle navedenih komandi ne prelazi u naredni red, i u 6. glavi gde su delovi koda samo boldovani i izlaze van okvira teksta. Dodavanje nekih novih redova i naprednije korišćenje Latex-a će to lako rešiti. Što se čitljivosti tiče, takođe paziti i na interpunkciju. Time podrazumevam stavljanje zareza ispred jer, pa, kao i slično, kao i nestavljanje nepotrebnih crtica(3) ili dve tačke ("komunikacioni kanal: je veza izmeu") između reči koji već čine celovitu rečenicu.

Da bi rad bio dosledniji i da ne bi bilo osetno da je pisalo više osoba, bilo bi dobro izvršiti neka usaglašavanja. Problematično je što se negde koristi kompajlira, negde kompilira, negde debageri integrisani u razvojno okruženje, a negde debageri integrisani u razvojnom okruženju itd. Treba uskladiti i stil, tako da ne bude negde tekst u vidu teza(4.3), a negde rečenica.

Još neke preporuke za popravku:

Trenutno	Savet
ne želite da promenite stanje programa na koji želite da ga pokrenete	za koji želite da ga pokrenete
Zbog toga je potreban veoma jak razlog zbog koga se ne mogu savladati problemi već postojeće rasprostranjene alate	postojećim rasprostranjenim alatima
sprečilo menjanje stanja programa ili da prouzrokuje beskonačnu petlju.	beskonačna petlja
Run-time greške ne registruje prevodilac ali može uzrokovati da računar abnormalno prekine sa radom u toku izvršavanja programa.	one mogu
tački prekida	tačaka prekida
nedostajanje	nedostatak
EBJ komponente su teške za debugovanje, razlog je taj što se	jer
lokacije tj stanja programa	stanja programa
Java debugging is a complex space prevedeno je kao Debugovanje Jave je kompleksan pojam	Možda bolje posao ili proces, s obzirom da se kasnije nastavlja priča u tom kontekstu.

4.4 Provera sadržajnosti i forme seminarskog rada

- Da li rad dobro odgovara na zadatu temu?
Rekla bih da da, jer je predstavljeno dosta mogućnosti koji se mogu koristiti za ovaj programski jezik.
- Da li je nešto važno propušteno?
Možda neka priča o izuzecima?
- Da li ima suštinskih grešaka i propusta?
Ne bih rekla, pokrivene su sve važne teme na korektan način, samo je potrebno sam tekst malo preformulisati.
- Da li je naslov rada dobro izabran?
Naslov je isti kao i tema. Nije menjano ništa. Pretpostavljam da je to u redu, s obzirom da je sveobuhvatan.
- Da li sažetak sadrži prave podatke o radu?
Ne sadrži motivaciju i opravdanje, kao ni cilj ili šta se postiže čitanjem rada. Suština i sam sadržaj rada su detaljno predstavljani.
- Da li je rad lak-težak za čitanje?
Neki delovi su lako čitljivi. Ovi koji nisu, mogu to postati malo boljim stilom, elaboracijom i ostvarivanjem malo smislenijih prelaza između rečenica.
- Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Poželjno je poznavanje Jave i njene virtuelne mašine, kao i nekog IDE-a

i programerskih termina kao što je ad hoc recimo, koji je nekoliko puta referisan u radu. Međutim, nije neophodno.

8. Da li je u radu navedena odgovarajuća literatura?
Ima ih pregršt. Knjiga koja je korišćena je poprilično stara(1998), nisam sigurna koliko je "moderna"i da li je bilo bolje koristiti nešto drugo. Verujem da tu uskaču brojni navedeni linkovi. Neki od njih su jako dobri.
9. Da li su u radu reference korektno navedene?
Nisu. Ne postoji nijedna. Mislim da nije korišćen bibtex, a nema ni "ručnih" referisanja na navedene izvore.
10. Da li je struktura rada adekvatna?
Prva strana sadrži sve što treba, osim ključnih reči. Prvi pasus se može smatrati uvodom, ali zaključak ne postoji. Imena celina bi mogla da se promene i drugačije rasporede da bi bio jasniji rad. Druga polovina je solidno organizovana. Mada mislim da bi bilo dobro celine organizovati i gradacijski, tako da se složenost metoda povećava kako se ide kroz rad, što ovde nije slučaj.
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)
Zadovoljen je broj strana. Ima ih 11. Takođe, postoje po jedna slika i tabela, kao i graf.
12. Da li su slike i tabele funkcionalne i adekvatne?
Slika se ne vidi, mada po njenom nazivu, rekla bih da je na adekvatnom mestu. Graf ima više grana i jako je veliki, što ima suprotan efekat na čitljivost od očekivanog. Tabela je malo šira.

4.5 Ocenite sebe

Rekla bih da sam malo upućena. Upoznata sam sa osnovnim konceptima, problemima i ciljevima debugovanja, jer imam programerskog iskustva. Međutim, što se tiče Java, nisam koristila ništa više nego što mi je bilo neophodno za potrebe kursa Objektno-orijentisano programiranje. Tu sam koristila debager samo par puta u manjim sistemima. Što se tiče naprednijih pomenutih sistema i tehnologija ovde, potpuni sam amater.

Glava 5

Dodatne izmene