Načini debagovanja u programskom jeziku Python

Dimitrije Sekulić, Sandra Radojević, Maja Gavrilović, Matija Pejić

15. april 2020

Sadržaj

Glava 1

Recenzent — ocena: 3

1.1 O čemu rad govori?

Rad obrađuje temu pronalaženja i otklanjanja grešaka (debagovanja) u *Python* jeziku. Prva polovina rada obrađuje opšti proces debagovanja i česte, jednostavne metode ispravljanja grešaka. U ovom delu su opisani koraci formalnog pristupa pronalaženja grešaka. U drugom delu rada su date instrukcije za pravilno korišćenje alata predviđenih za debagovanje *Python* koda.

1.2 Krupne primedbe i sugestije

Poželjno je preformulisati sve rečenice napisane u prvom licu množine u neutralni oblik. Na primer, predlaže se zamena rečenice:

„Proces debagovanja nam omogućava pronalaženje grešaka i rešavanje ostalih problema unutar programa.“

rečenicom:

„Debagovanje je postupak pronalaženja grešaka i rešavanja drugih problema koji se javljaju prilikom izvršavanja programa.“ Recenzija je prihvaćena u ovoj rečenici s obzirom da je stil isti i da se ne menja, međutim smatramo da je u nekim delovima rada u redu da se koristi prvo lice množine s obzirom da čitalac vodi kroz primere (Poglavlje 6, PDB debager).

1.2.1 Sažetak

1. Prva rečenica govori šta proces debagovanja omogućava umesto da opiše šta je debagovanje. Dat je predlog alternative za prvu rečenicu. Malo pre je prihvaćeno sa obrazloženjem.

2. U trećoj rečenici od kraja sažetka, koja počinje sa „Izložićemo...“ je napisano na kraju „prigodne uslove za njihovo korišćenje“. Nije jasno potpuno šta su uslovi za korišćenje tehnika. Poželjno je detaljnije objasniti taj segment rečenice. Nije prihvaćeno. Smatramo da u sažetku ne treba da bude nikakvo objašnjenje. Čitalac se poziva da vidi šta su uslovi u tekstu. Na primer, u 7.1 pišemo da koristimo detaljno debagovanje u PyCharmu ako želimo da vidimo izvršavanje korak po korak, što je uslov za korišćenje te metode.

3. Sledeća rečenica, počinje sa „Da bi se postigao...“ ne objašnjava šta je urađeno u radu, samo je napisana kao činjenica. Nije jasno šta su „tehnike i alati za brzo uočavanje propusta“ jer to nije zadato u opisu debagera. Preformulisati rečenicu tako da opisuje šta je u radu urađeno. Nije prihvaćeno. Rečenica jeste opštija, ali sadrži suštinu rada. Smatramo da ne treba ovde nabrajati i da se u sadržaju ispod sažetka može videti šta se u radu tačno nalazi od tehnika.

4. Poslednja rečenica sažetka nije jasna, i ne opisuje šta je postignuto u radu. Sama po sebi nije jasna i šta i zašto je potrebno da se izdvoji iz mora sličnih. Sugeriše se razjašnjenje ili izbacivanje rečenice. Prihvatamo sugestiju. S obzirom da u sažetku treba navesti i razlog za čitanje rada, mi smo ovde kao razlog naveli unapređenje programerskih veština.

1.2.2 Uvod

1. S obzirom da su kasnije korišćene reči debagovanje i debager bez zadavanja njihovih definicija ili detaljnijih objašnjenja, u uvodu bi bilo dobro na početku definisati ta dva pojma. Prihvatamo sugestiju. Izmenjeno je i delovi iz prvog naslova posle uvoda su prebačeni u uvod. Takođe smo reformulisali početak prve sekcije.

2. Poslednja rečenica uvoda objašnjava tok celog rada, što je redundantno, jer je tok rada vidljiv iz sadržaja. Umesto toga bi pre pretposlednje rečenice bilo korisno opisati detaljnije zamisao automatizovanog debagera. Prihvaćena sugestija. Rečenica je izbačena i taj deo je reformulisan.

1.2.3 Sintaksne greške u Python-u

1. Četvrta rečenica sekcije Izuzeci u *Python*-u, nije potpuno jasna, nije jasno zašto je (i da li je uopšte) korisno što su izuzeci bagovi za koje znamo da postoje. Predlog izmene je zameniti rečenicu sledećom: „Pošto su izuzeci jasno vidljivi pri kompajliranju, mogu da se lako prepoznaju, pa stoga i uklone.“ Sugestija prihvaćena. Rečenica zamenjena drugom pogodnijom rečenicom „Ako u programu dođe do izuzetka...“

1.2.4 Semantičke greške u Python-u

1. Pre trećeg primera bi trebalo da se stavi rečenica „U primeru 3 je definisan program...“ umesto da posle trećeg primera piše „U prethodnom primeru...“. Sugestija prihvaćena. Dodata referenca iz teksta na primer.

2. Poslednja rečenica segmenta Semantičke greške u *Python-u* je nepotrebna jer prepričava sledeći deo rada. Uvod segmenta Debagovanje naučnom metodom dovoljno povezuje ova dva dela rada. Primedba prihvaćena. Rečenica je izbačena.

1.2.5 Debagovanje naučnom metodom

1. Naslov segmenta je neprikladan jer se stvara utisak da je naučna metoda jedna vrsta debagovanja a ne opšte pravilo sistematičnog pristupa

problemu. Predlozi za izmenu naslova su: „Primena naučne metode“ ili „Naučni pristup debagovanju“. Prihvaćena primedba. Prihvaćen drugi predlog za naslov.

2. Prva rečenica u ovom segmentu koja odgovara na pitanja, počinje sa „Tada se treba okrenuti...“ ima isti problem kao i naslov. Stvara utisak da postoji više formalnih načnina nalaženja problema a naučna metoda je jedna od njih. Zajedno sa sledećom rečenicom deluje kao da je naučni metod poseban pristup debagovanju. Poželjno je preformulisati taj segment tako da bude jasno da je naučni metod generalni naučni pristup problemu i da je koristan i za pronalaženje greške u kodu. Predlog formulacije:

   „Potrebno je sistematski analizirati izvršavanje programa kako bi se pronašao uzrok semantičke (ili neke druge) greške. Primena naučnog metoda postavljanja i testiranja hipoteze se pokazala kao dobra tehnika i u procesu debagovanja.“ Delimično prihvaćeno. Izbacili smo reč zbog koje je delovalo da ima više metoda i sada se vidi da je u pitanju tačno jedna. Predložena reformulacija nije prihvaćena, jer smatramo da se ne uklapa uz prethodne rečenice.

3. Sledeća rečenica „On traženje greške bazira...“ mislim da nije dobro prevedena, jer *framework* ovde ne predstavlja okruženje nego okvir. Nije jasno kako se prikupljaju dokazi za greške u programu. Druga polovina rečenice koja se odnosi na testiranje i održavanje koda nije potrebna jer je u sledećem pasusu to ponovo detaljnije objašnjeno. Predlog izmene rečenice:

   „Naučni metod je opšti koncept uz koji je moguće uskladiti druge, detaljnije metode.“ Delimično prihvaćena. Promenjena reč okruženje na reč okvir. Ostale primedbe su stal stila.

4. U poslednjoj rečenici na četvrtoj strani je zbunjujuće šta je reprodukcija greške. U knjizi Rotera K. je opsežno objašnjeno šta je reprodukcija greške i bilo bi korisno to objasniti u jednoj rečenici. Nije prihvaćena. Nismo želeli da proširujemo teskt toliko na tom mestu, jer to nije cilj rada i opsežno je. Nije nešto što se može navesti u jednoj rečenici i zato smo dodali dodali referncu na knjigu.

1.2.6 Debagovanje print naredbama

1. Prva rečenica je loša kao uvodna u ovaj segment. Umesto toga ukratko opisati kako se debaguje pomoću *print* naredbe, a nju potpuno izbaciti. Nije prihvaćeno. Nije objašnjeno zašto je ova rečenica loša, predstavlja dobru osnovu za ostatak teksta. Nije moguće objasniti u jednoj rečenici za šta služi cela glava.

2. U trećoj rečenici, „Iako jednostavan i nedvosmislen, to ne znači da je bez greške.“ „to ne znači da je bez greške“ stilski preformulisati. Na primer napisati: „Iako je ovaj pristup jednostavan i nedvosmislen ima i mane.“ Nije prihvaćeno. Promena ne doprinosi unapređenju rada.

3. Sledeća rečenica je stilski zbunjujuća. Nije jasno na šta se odnosi pridev „veći“. Objasniti da li remeti eleganciju ako je kod duži ili ispis duži. Prihvaćena primedba. Promenjeno da piše „duži“.

4. U trećem pasusu postoji citat slikovitog poređenja za koji u tekstu nije objašnjeno da je citat, pa stoga odskače od ostatka teksta. Napisati uvodnu rečenicu da je u knjizi Rothera K. dato navedeno poređenje. Nije prihvaćeno. Nismo hteli da menjamo dosadašnji stil citiranja. Pored toga smo primetili da na tom mestu nije citiran pravi izvor i to je promenjeno.

1.2.7 Debagovanje u okruženju PyCharm

1. Sve što je rečeno u prvoj rečenici je naslovom već rečeno. Umesto toga bi bilo korisnije u jednoj ili dve rečenice u najkraćim crtama opisati šta je *PyCharm*. Sugestije prihvaćena. Rečenica je zamenjena sa objašnjenjem šta je *PyCharm*.

2. Ni za jednu sliku u segmentu Debagovanje u okruženju *PyCharm* nije dat naslov slici niti je negde u tekstu data referenca na sliku. Sugestija prihvaćena. Dodati opisi slika i reference na njih.

1.3 Sitne primedbe

Veliki broj reči na engleskom jeziku nisu napisane italik fontom. Svaki naziv funkcije, engleski naziv i naziv jezika *Python* napisati italik fontom. Delimično je prihvaćeno. Rad ima previše engleskih reči i bilo bi nepregledno kad bi sve bile italik fontom. Recimo, bilo bi nečitljivo kad bi svaka reč Python bila italik s obzirom da se često koristi.

1.3.1 Sintaksne greške u Python-u

1. Prva rečenica podsekcije Izuzeci u *Python*-u, „zajedno čine hijerarhiju“ predložena izmena reči „čine“ u „formiraju“. Prihvaćeno iako je značenje isto. Ova ispravka ne dodaje nikakvo značenje rečenici.

2. Treća rečenica podsekcije Izuzeci u *Python*-u je citat iz knjige Rotera K. (referenca 8), bilo bi poželjno istaći da su u toj knjizi izuzeci slikovito objašnjeni tom rečenicom a zatim je citirati, kako bi se skladnije uklopila u ostatak teksta. Nije prihvaćeno. Kao što smo već naveli nismo hteli da menjamo dosadašnji stil citiranja.

3. Rečenica pre primera 1 u sekciji Čitanje koda na mestu bag ne pravi referencu na primer 1, bilo bi poželjno napraviti referencu. Ista primedba za sledeću rečenicu napisanu između dva primera koja nema referencu na drugi primer. Sugestija prihvaćena. Dodate su reference iz teksta na primer.

4. U opisu primera 2 staviti da je on ispis iz konzole za prvi primer, a ne za prethodni. Sugestija prihvaćena. Dodate su reference iz teksta na primer.

5. U delu Poruka o grešci, na početku drugog pasusa nema potrebe napisati „Tip greške jeste...“ dovoljno je napisati „Tip greške je...“. Sugestija je prihvaćena.

6. U odeljku Hvatanje izuzetaka u drugoj rečenici od pozadi, u prvom pasusu je reč uhvatiti napisana pod navodnicima posle kojih nema razmaka, oba navodnika su gornja (Moguće je kopirati *unicode* donje i gornje navodnike i staviti ih tako u tekst). Sugestija prihvaćena. Promenjeni su navodnici.

1.3.2 Semantičke greške u Python-u

1. Pre poslednje rečenice na trećoj strani je verovatno napisan dupli razmak. U LaTeX-u je jedan razmak.

1.3.3 Debagovanje naučnom metodom

1. Pretposlednja rečenica na 4. strani, „Neko nepisano pravilo“ zameniti sa „Opšti savet“. Prihvaćena sugestija. Promenjeno da piše „Opšti savet“.

2. U prvoj rečenici na 5. strani objasniti šta je „nekakva nasumičnost“ ili izbaciti taj deo. Prihvaćeno. Taj deo je izbačen.

3. U poslednjoj rečenici segmenta Debagovanje naučnim metodom umesto „ćemo pričati kasnije“ napisati „će biti reči kasnije“. Sugestija delimično prihvaćena. Taj deo je izbačen i reformulisan, zaključili smo da je taj deo suvišan.

1.3.4 Debagovanje print naredbama

1. Referenca data posle citata iz knjige na kraju trećeg pasusa vodi ka sajtu *Python* dokumentaciji, a ne ka knjizi iz koje je preuzeta. Prepraviti na referencu 8. Ispravljeno.

2. U poslednjem pasusu uvoda segmenta Debagovanje print naredbama je iskorišćena fraza „isforsiramo ispis“ koja deluje previse neformalno. Zameniti je nekom formalnijom. Nije prihvaćena. Isforsirati najbolje opisuje šta se dešava i pozajmljenica je iz engleskog jezika.

1.3.5 PDB debager

1. U prvoj rečenici pre zagrade nedostaje razmak. Izmenjeno je.

2. Rečenica „Zato se u ovom delu upoznajemo sa njim.“ je nepotrebna. Prihvaćeno. Rečenica je izbačena.

3. Na kraju strane 6. „Koristeći pip većinu verzija“ je loše frazirano. Ako je moguće koristiti većinu verzija *pip*-a onda je dovoljno reći samo *pip*. S obzirom da se u ovoj rečenici prvi put pojavljuje reč *pip*, bilo bi korisno napisati *pip* sistem za kontrolisanje paketa, ili staviti fusnotu koja objašnjava šta je *pip*. Prihvaćena je primedba. Uklonjen je deo teksta koji se odnosi na instaliranje ipdb debagera, s obzirom da on nije ni korišćen dalje u primerima.

4. Na 10. strani je naziv tabele stavljen iznad tabele a za sve ostale slike i primere je naslov stavljen ispod. Nije prihvaćeno. Po materijalima sa časa naziv i opis tabele se stavlja iznad tabele.

1.3.6 Debagovanje u okruženju PyCharm

1. Na 12. strani, u poslednjoj rečenici dela Debagovanje u okruženju *PyCharm* je iskorišćena reč „jako“ nepravilno, pravilno bi bilo napisati „veoma“. Usvojeno i promenjeno.

2. U istoj rečenici su korišćeni navodnici posle kojih se ne pravi razmak, oba navodnika su gornja. Usvojeno i prihvaćeno.

1.3.7 Zaključak

1. U prvoj rečenici je nepravilno iskorišćena reč „jako“, dovoljno je samo reći „*Python* je popularan jezik“. Reč „apstraktnijem“ zameniti „apstraktnom“. Sugestija delimično prihvaćena. Promenjeno na popularan. Druga primedba nije izmenjena, jer smatramo da je i ovako ispravan tekst.

2. Druga rečenica nije potrebna, preporučuje se da se izbaci. Nije prihvaćeno, neophodno je zbog naredne rečenice.

3. U trećoj rečenici „alate za debagovanje“ zameniti sa „debagere“. Reč „posrednici“ zameniti sa „alati“. Umesto „čineći taj proces“ dovoljno je napisati „čineći proces“. „Nenapornim“ je stilski čudna konstrukcija, napisanti „manje napornim“ ili „manje opterećujućim“. Delimično prihvaćena primedba. Nenaporni promenjen u manje naporni.

4. U četvrtoj rečenici umesto „tako imamo odličnu kombinaciju“ napisati „tako dolazimo do odlične kombinacije“. Nije prihvaćeno. Smatramo da nije rečenica dvosmislena i da ovakva promena ne menja kontekst.

5. U petoj rečenici nije potrebno da se kaže čitaocu šta mu je sada jasnije. Rečenicu preformulisati tako da se izbaci taj deo. Nije prihvaćeno. Iskorišćen je izraz poznato, a ne jasnije. Ako je čitalac pročitao rad njemu jesu poznati pomenuti mehanizmi.

6. Poslednja reč zaključka je „preferensi“, što je netačan oblik. Zameniti sa „preferenci“. Prihvata se.

1.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
   Rad dobro odgovara na zadatu temu. Koncept rada je veoma dobro napravljen. Sviđa mi se što su prvo detaljno opisani metodi generalnog debagovanja programa i što je opisan naučni pristup pronalaženja grešaka, a kasnije opisani postupci debagovanja u *Python*-u.

2. Da li je nešto važno propušteno?
   Ne smatram da je išta važno propušteno.

3. Da li ima suštinskih grešaka i propusta?
   Nema suštinskih grešaka i propusta.

4. Da li je naslov rada dobro izabran?
   Naslov rada je jednostavan za razumevanje i odgovara zadatoj temi.

5. Da li sažetak sadrži prave podatke o radu?
   U sažetku nije navedeno ništa što nije obrađeno u radu, ali deo sažetka uopšte ne opisuje o čemu rad govori, nego daje činjenice. Dati su predlozi kako izmeniti sažetak.

6. Da li je rad lak-težak za čitanje?
   Rad je napisan stilom jednostavnim za čitanje.

7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
   Potrebno je predznanje za čitanje rada. Potrebno je poznavanje *pip*-a i *PyCharm* okruženja s obzirom da nisu data objašnjenja šta oni predstavljaju. Ako bi autori dodali ukratko opise ovih alata, bilo bi dovoljno opšte poznavanje pisanja *Python* programa za čitanje ovog rada. Smatramo da ne treba da posvetimo toliki fokus na samo korišćenje pip alata i PyCharm okruženja. Bilo kakav opis ne može da zameni pravo znanje alata. Ovde se kreće od početka i elementarnih stvari. U tekstu ne koristimo nikakve napredne koncepte i mogućnosti tih alata.

8. Da li je u radu navedena odgovarajuća literatura?
   Sva litaratura korišćena u radu je navedena korektno.

9. Da li su u radu reference korektno navedene?
   Reference jesu korektno navedene, sve što je preuzeto sa drugog izvora je referencirano.

10. Da li je struktura rada adekvatna?
    Struktura rada se uklapa u sva propisana pravila.

11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
    Rad sadrži sve elemente propisane uslovima, sadrži slike, tabelu i uklapa se u predviđeni broj strana. Jedina zamerka vezana za ove detalje je što većina slika nije referencirana u samom tekstu i u sedmom odeljku ni jednoj slici nije dat opis. Zamerka popravljena.

12. Da li su slike i tabele funkcionalne i adekvatne?
    Tabele i slike jesu adekvatne i korisne.

1.5 Ocenite sebe

Srednje sam upućena u ovu oblast.

Nisam debagova programe pisane u jeziku *Python*, tako da ne smatram da sam previše upućena, ali jesam pregledala svu navedenu literaturu koju su autori koristili u radu i izučila detaljno poglavlja koja su citirali.

Glava 2

Recenzent — ocena: 4

2.1 O čemu rad govori?

Na početku se govori o debagovanju u opštem smislu, a zatim se prelazi na tipove grešaka u Python-u. Tu se zapaža da su klase grešaka u Python-u izvedene iz osnovne klase Exception. Nakon toga se prelazi na tipove debagovanja, a zatim se pažnja usmerava na karakteristike i objašnjenje opcija i toka u debagovanju pdb debagerom i debagerom u okruženju PyCharm.

2.2 Krupne primedbe i sugestije

U zaključku stoji da se ostavlja čitaocu da izabere odgovarajući metod. Međutim, nisu dovoljno objašnjene situacije u kojima određeni debageri dolaze do izražaja, kao i prednosti korišćenja jednog u odnosu na neki drugi. Korisno bi bilo napisati i njihove mane, tj. situacije u kojima nisu od pomoći.
U prvoj polovini rada za pomenute tehnike pomenute su prednosti i mane. U drugoj polovini obrađeni alati su celokupni i potpuno ravnopravni i čiji izbor zavisi od preferenci korisnika. Nije neophodno međusobno porediti metode, ako smo već naveli za svaku metodu ponaosob prednosti i mane.
U sekciji 2 naslov Sintaksne greške u Python-u nema baš prevelike veze sa objašnjenjem u pasusu - nigde nema reči o sintaksi programskog jezika Python kao i njihovim greškama, govori se o debagovanju u opštem smislu.
Sugestija prihvaćena. Izmenjeni su naslovi i reformulisan je uvodni deo.
Takođe, u sekciji 7.1, za recenzenta je malo nejasno objašnjenje opcije Step into my code. Koliko je meni poznata ta opcija, trebalo je napisati da neće ući u bibliotečku funkciju, ali hoće u funkciju koja je definisana u našem kodu. Funkcija f, koja je data kao primer, pretpostavljam da je definisana u našem kodu, tako da će se korišćenjem opcije Step into my code ući u tu funkciju, što se kosi sa objašnjenjem. Ako je to neka bibliotečka funkcija onda je trebalo to naglasiti kao što je to naglašeno u slučaju funkcije random.nextInt().
Sugestija prihvaćena. Pasus izmenjen i pojašnjeno je šta se dešava.

2.3 Sitne primedbe

- U sekciji 2.4 štamparska greška - umesto loše putanju trebalo je napisati lošu putanju Sugestija nije prihvaćena. Ovde se loše koristi kao prilog, a ne kao pridev.
- U sekciji 4 štamparska greška - umesto trebao trebalo je napisati trebalo Sugestija prihvaćena.
- U istoj sekciji štamparska greška - umesto sitematičnu trebalo je napisati sistematičnu Sugestija popravljena.
- U sekciji 7.1 štamparska greška - umesto predhodnog trebalo je napisati prethodnog Sugestija popravljena.
- U sekciji 7.4 štamparska greška - umesto vestackinačin trebalo je napisati "veštački"način Prihvaćeno sugestijom prethodnog recezenta.

2.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
   Rad dobro odgovara na zadatu temu, jer daje odgovore na sva ključna pitanja.
2. Da li je nešto važno propušteno?
   U radu su obrađene sve ključne teme.
3. Da li ima suštinskih grešaka i propusta?
   U principu ne. Sugestije recenzenta su date u sekciji Krupne primedbe i sugestije.
4. Da li je naslov rada dobro izabran?
   Tekst rada je u skladu sa naslovom, tako da jeste.
5. Da li sažetak sadrži prave podatke o radu?
   U sažetku se nalazi kratak opis o svim temama obrađenim u radu.
6. Da li je rad lak-težak za čitanje?
   Rad je lak za čitanje, kao i razumevanje jer ima dosta primera.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
   Za razumevanje teksta je korisno imati osnovno predznanje o programskom jeziku Python.
8. Da li je u radu navedena odgovarajuća literatura?
   U radu je navedena odgovarajuća literatura, što se odnosi i na strukturu literature (bar jedna knjiga, bar jedan naučni članak, bar jedna adekvatna veb adresa).
9. Da li su u radu reference korektno navedene?
   U radu su označene i korektno navedene reference, u tekstu je označeno odakle su informacije pronađene.

10. Da li je struktura rada adekvatna?
    Rad je adekvatno i uredno struktuiran.

11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
    Svi elementi propisani uslovom se nalaze u radu - sadrži originalnu sliku i tabelu, a dužina je optimalnih 12 strana, kao i uslovi vezani za literaturu.

12. Da li su slike i tabele funkcionalne i adekvatne?
    Slike i tabele opisuju tekst uz koji se nalaze.

2.5 Ocenite sebe

Kao student Matematičkog fakulteta zbog čega imam predznanje iz Python-a, a sa druge strane autor rada čija je tema takođe vezana za debagovanje mislim da sam veoma upućen u ovu oblast.

Glava 3

Recenzent — ocena: 3

3.1 O čemu rad govori?

Ovaj rad sadrži opise dva tipa grešaka, Sintaksne i Semantičke i više metoda kojima se one mogu razrešiti. U zavisnosti od metode u pitanju, opis nje stavlja akcenat na osobine te metode, praktičnu primenu ili teorijsku suštinu iza nje.

3.2 Krupne primedbe i sugestije

1. Početak sekcije sa sintaksnim greškama bi trebao da se odradi ispočetka. Postojeći tekst je prikladniji za Uvod. Sugestija prihvaćena na osnovu prethodnih recezenata.

2. Podsekcije o poruci greške i o hvatanju izuzetaka nisu za sekciju o Sintaksnim greškama. One su više za posebnu sekciju, imajući u obzir da se pojavljuju uz oba tipa grešaka. Sugestija prihvaćena na osnovu prethodnih recezenata.

3. Sekcija o naučnoj metodi u nekim delovima direktno kopira tekst iz Izvora 8. Opis naučne metode, na primer, je direktan prevod. Iako je odlično napisan deo, čitanje izvora pokazuje koliko je intenzivno vršeno izvlačenje teksta, čak i direktno kopiranje. Autor mora da prepravi tekst, u ovom stanju on je napisao plagijat. Nismo sigurni na šta recezent misli. Koraci jesu preuzeti iz knjige na koje jeste navedena pogodna referenca. Sve ostalo smatramo da nije kopirano, jer i nije. Autori su se zaista dovoljno potrudili da prepričaju suštinu knjige svojim rečima, iako je ovaj deo pisan iz samo jednog izvora.

4. Sekcija o Pycharm debagovanju stavlja preveliki fokus na "šta koje dugme radi". Skratiti taj deo, dodati praktičan primer i neke prednosti i mane metoda. Pogledati još neku dokumentaciju ako se može naići na još informacija. Primedba nije prihvaćena. Debagovanje u PyCharm je praktična stvar i mora da se prikaže šta koje dugme radi. Smatramo da ako bi samo opisno opisali dugmiće ne bi bilo dovoljno jasno.

3.3 Sitne primedbe

1. U nekim sekcijama se nailazi na tekst koji je suviše neformalan. Na primer, "Neko nepisano pravilo je da ga primenimo ako ne nademo rešenje u 10-15 minuta.". Ovaj stil je ok za laički priručnik, ali ne za seminarski rad. Prihvaćeno po savetima prvog recezenta.

2. Neke sekcije intenzivno koriste naučne/tehničke termine uz slabo objašnjavanje. To dovodi do otežavanja razumljivosti rada za one koji nisu upoznati. Predlažem korišćenje fusnota ili zamenu određenih termina laičkim terminima s istim značenjem. Sugestija prihvaćena. Po savetima prethodnih recezenata su izmenjene neke reči, ali nismo izmenili neke delove gde bi se time narušila razumljivost i prevod bi delovao isforsirano.

3. Proći kroz rad par puta i ispraviti sintaksne greške. Prihvaćena sugestija. Rad je pročitan više puta. I uz pomoć prethodnih recezenata ispravljene štamparske greške.

3.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
   Da. Rad kao celina je dobro konstruisan, lepo napisan i svaka metoda je fino objašnjena.

2. Da li je nešto važno propušteno?
   Ne. Rad je dobro dizajniran, dobre su metode izabrane i svaka od njih je barem suštinski opisana.

3. Da li ima suštinskih grešaka i propusta?
   Ne. Autori su za svaku sekciju razumeli teme koje obrađuju i dobro napisali.

4. Da li je naslov rada dobro izabran?
   Da. Suština rada jeste u opisima različitih metoda debagovanja u Pythonu, otud je naslov skroz adekvatan i pritom dobro izabran.

5. Da li sažetak sadrži prave podatke o radu?
   Da. Sadrži sve potrebne informacije i nudi motivaciju za pravljenje rada.

6. Da li je rad lak-težak za čitanje?
   U određenim sekcijama nema dovoljno objašnjenja i previše se stavlja akcenat na tehničke/naučne termine, ali kao celina je lak za čitanje.

7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
   Smatram da će svako sa osnovnim znanjem Pythona moći da razume tekst bez većih problema.

8. Da li je u radu navedena odgovarajuća literatura?
   Jeste. Analiza literature pokazuje da su sve informacije iz rada izvučene iz nje.

9. Da li su u radu reference korektno navedene?
   Jesu.

10. Da li je struktura rada adekvatna?
    Jeste. Ovaj rad je rezultat dobre primene "top-down, inside-out"pristupa.

11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
    Da. Primenjuju se slike i tabele i dovoljno je dugačak.

12. Da li su slike i tabele funkcionalne i adekvatne?
    Jesu. Pogotovo tabela o PDB funkcionalnostima i slika Pycharm Debug dugmića.

3.5 Ocenite sebe

Srednje sam upućen u oblast. Na poslu redovno koristim Python i koristio sam sve metode navedene ovde. Nisam ulazio posebno u dokumentaciju iza njih izvan konkretnih potreba zadatka.

Glava 4

Dodatne izmene

Izmenjene su sintaksne greške u sažetku i zaključku kao i u pojedinim naslovima koje recezenti nisu primetili.