

## ПРИРОДНО-МАТЕМАТИЧКИ ФАКУЛТЕТ

## ИЗВЕШТАЈ О ОЦЕНИ ДОКТОРСKE ДИСЕРТАЦИЈЕ

<b>I ПОДАЦИ О КОМИСИЈИ</b>
<p>1. Датум и орган који је именовao комисију</p> <p>Дана 20.12.2018. године Наставно-научно веће Природно-математичког факултета у Новом Саду је именовало Комисију за оцену и одбрану докторске дисертације.</p> <p>2. Састав комисије са назнаком имена и презимена сваког члана, звања, назива уже научне области за коју је изабран у звање, датума избора у звање и назив факултета, установе у којој је члан комисије запослен:</p> <ol style="list-style-type: none"> <li>1. др Милош Радовановић, ванредни професор, рачунарске науке, 12.12.2016, Универзитет у Новом Саду, Природно-математички факултет, председник</li> <li>2. проф. др Зоран Будимац, редовни професор, рачунарске науке и информатика, 15.06.2004. Универзитет у Новом Саду, Природно-математички факултет, ментор</li> <li>3. др Гордана Ракић, доцент, рачунарске науке, 10.02.2016. Универзитет у Новом Саду, Природно-математички факултет, члан</li> <li>4. др Захарије Радивојевић, ванредни професор, рачунарска техника и информатика, 04.02.2018. Универзитет у Београду, Електротехнички факултет, члан</li> </ol>
<b>II ПОДАЦИ О КАНДИДАТУ</b>
<ol style="list-style-type: none"> <li>1. Име, име једног родитеља, презиме: Дони, Александар, Працнер</li> <li>2. Датум рођења, општина, држава: 04.12.1984, Нови Сад, Република Србија</li> <li>3. Назив факултета, назив студијског програма дипломских академских студија – мастер и стечени стручни назив Универзитет у Новом Саду, Природно-математички факултет, информатика, мастер информатичар</li> <li>4. Година уписа на докторске студије и назив студијског програма докторских студија Уписан 2014. године на студијски програм: Докторске академске студије информатике</li> <li>5. Назив факултета, назив магистарске тезе, научна област и датум одбране: -</li> <li>6. Научна област из које је стечено академско звање магистра наука: -</li> </ol>
<b>III НАСЛОВ ДОКТОРСKE ДИСЕРТАЦИЈЕ:</b>
<p><b>Translation and Transformation of Low Level Programs</b> <b>Превођење и трансформисање програма ниског нивоа</b></p>

#### IV ПРЕГЛЕД ДОКТОРСKE ДИСЕРТАЦИЈЕ:

Докторска дисертација садржи укупно 211 страна, 9 поглавља, 4 прилога, 87 библиографских референци, 23 табеле и 36 слика.

Дисертација се бави проблемима превођења и реструктурирања програма писаних у језицима ниског нивоа. Као решење је дат процес у коме се из почетног језика ниског нивоа програм преводи у језик WSL у оквиру кога је могуће примењивати трансформације које очувавају семантику. Адекватном применом трансформација се логика програма може подићи на виши ниво и тиме омогућити лакше разумевање и одржавање софтвера. У оквиру дисертације је описан аутоматизован алат за одабир и примену трансформација са овим циљем.

Дисертација је структурирана на следећи начин:

##### Part I: Preliminaries (Део I: Предзнање)

Chapter 1: Introduction (Поглавље 1: Увод)

Chapter 2: Software Maintenance, Evolution and Metrics (Поглавље 2: Одржавање софтвера, еволуција и метрике)

Chapter 3: FermaT and WSL (Поглавље 3: FermaT и WSL)

Chapter 4: Assembly Language (Поглавље 4: Асемблерски језик)

Chapter 5: Bytecode and MicroJava Language (Поглавље 5: Бајткод и језик Микро Јава)

##### Part II: Translation (Део II: Превођење)

Chapter 6: Translating Assembly Code (Поглавље 6: Превођење асемблерског кода)

Chapter 7: Translating MicroJava Bytecode (Поглавље 7: Превођење Микро Јава бајткода)

##### Part III: Transformation (Део III: Трансформација)

Chapter 8: Code Transformations (Поглавље 8: Трансформације кода)

Chapter 9: Thesis Conclusions (Поглавље 9: Закључци дисертације)

##### Part IV: Appendices (Део IV: Прилози)

Appendix A: FermaT Transformations Catalogue (Прилог A: Каталог FermaT трансформација)

Appendix B: MicroJava Specifics (Прилог B: Детаљи Микро Јаве)

Appendix C: Additional Data (Прилог C: Додатни подаци)

Appendix D: Source Code (Прилог D: Изворни код)

Део I садржи пет поглавља са предзнањем неопходним за разумевање остатка дисертације.

У првом поглављу је дат кратак преглед проблема који се решава и урађеног решења. Потом су експлицитно излистани доприноси дисертације.

Друго поглавље представља преглед основних појмова еволуције софтвера и класичних проблема у одржавању софтвера. На крају главе су укратко описане софтверске метрике и њихова примена.

Треће поглавље описује програмски (и не само програмски) језик WSL (енг. *Wide Spectrum Language* – језик широког спектра) и систем за трансформације FermaT који је тренутна имплементација тог језика. Описане су теоријске основе језика, те како се могу креирати трансформације кода у оквиру њега, што је касније коришћено у дисертацији.

Поглавље четири је преглед асемблерског језика ниског нивоа, тачније Микрософт асемблера (MASM) за процесоре са x86 архитектуром. Представљене су основне особине процесора, као и инструкција које су подржане на њему, те неки од концепата директно везаних са сам асемблерски језик.

Поглавље пет представља појмове бајткода, као извршне верзије програма, али намењене за неку специфичну виртуелну машину (нпр. Јава виртуелна машина), а не директно за неки процесор. Даље је представљен програмски језик Микро Јава, који је подскуп језика Јава, те је представљена виртуелна машина за тај језик.

Део II се састоји из два поглавља која описују поступак превођења програма написаних у језицима ниског нивоа.

Поглавље шест приказује развијени алат за превођење из представљеног асемблера у језик WSL.

Поглавље седам приказује други развијени алат за превођење бајткода добијеног од Микро Јава програма у WSL.

Део III је најзначајнији део дисертације, и садржи једно поглавље, поглавље осам, које се бави трансформацијама кода. На почетку је објашњен поступак ручног одабира трансформација и представљен је један детаљан пример тог процеса. Потом су представљене неке могућности за аутоматизацију тог процеса, те и један конкретан алат, развијен у оквиру дисертације, који то омогућава коришћењем алгоритма претраживања успињањем (енг. *hill climbing*). Поглавље се даље наставља примерима експеримената превођења и трансформисања са скуповима програма у асемблеру и Микро Јави.

Последње, девето поглавље представља преглед целокупног учинка и закључака у оквиру дисертације, као и поређење са неким другим приступима. На крају су дате могућности за даље развоје представљених алата и процеса.

Део IV садржи четири прилога који употпуњују дисертацију: каталог трансформација имплементираних у алату FermaT, опис језика Микро Јава и одговарајуће виртуелне машине, додатне табеле софтверских метрика израчунатих на Микро Јава програмима, и изворни код програма `hill_climbing`.

## V ВРЕДНОВАЊЕ ПОЈЕДИНИХ ДЕЛОВА ДОКТОРСКЕ ДИСЕРТАЦИЈЕ:

Дисертација је добро структурирана, како у целини, тако и по деловима. Дати су прегледи постојећих проблема и неких решења у одржавању и еволуцији софтвера. Представљени резултати и нови алати су допринос области рачунарских наука. Дисертација самим тим задовољава постављене циљеве.

**Наслов.** Наслов дисертације је јасно, прецизно формулисан и одражава садржај истраживања.

Први, уводни део дисертације представља неке од основа у области истраживања, почевши од еволуције и одржавања софтвера. Потом даје детаљнији приказ система FermaT и језика WSL, те пример имплементације једне нове трансформације. Објашњене су основе Микрософт асемблера за x86 процесор. Објашњене су предности виртуелних машина и приказана је конкретна машина за језик Микро Јава. Такође, описана су унапређења система FermaT имплементирана у оквиру рада на дисертацији: додате су и унапређене трансформације, унапређен поједностављивач израза, додат нови „тихи“ мод рада, као и нов начин прослеђивања параметара из командне линије.

У другом делу дисертације су приказана два преводиоца из језика ниског нивоа у језик WSL. Први преводилац прима програме који су подскуп асемблера за x86 процесор. Други ради са преведеним Микро Јава програмима, тј. са одговарајућим бајткодом. Описани су детаљи преводјења на неколико специфичних примера различитих типова операција. Код оба алата се као излаз добијају програми ниског нивоа у језику WSL, али постоје разлике у особинама генерисаних програма.

Трећи део дисертације се фокусира на сам процес трансформисања и подизања нивоа апстракције програма. Дат је пример ручног одабира и примене трансформација као илустрација процеса, али и потребе да се познаје систем да би се поступак успешно завршио. Потом су представљене неке од идеја како се процес може аутоматизовати. Дат је конкретан програм `hill_climbing` написан у језику WSL који користи алгоритам претраживања успињањем и структуралну метрику као функцију погодности (енг. *fitness function*) за добијање нових повољнијих програма. Приказани су примери конкретних трансформација неколико типова улазних програма. Дате су и анализе успешности процеса над скуповима програма добијених из асемблера и из Микро Јаве. Програми добијени из Микро Јаве су преводјени на различите начине, користећи три прекидача која постоје у алату и тиме је добијено осам варијанти програма. Над свим варијантама је коришћен исти програм за трансформисање и упоређене су разлике у крајњим резултатима. Ово даје бољи увид у функционисање процеса, као и какви програми су најпогоднији за тренутну верзију програма `hill_climbing`. Дате су и анализе времена извршавања програма `hill_climbing` у односу на метрике улазних програма. Уочено је да не постоје директне релације, односно да се не може јасно на основу датих метрика проценити трајање процеса. Постоји опште очекивање да ће дужи програми резултовати дужим трајањем трансформација, али није јасно колико дуже. Код асемблерских улаза није уочен јасан тренд раста, док код Микро Јава програма општи тренд раста може да се уочи, али постоји већи број изузетака, где програми са сличним метрикама имају значајно различита времена трансформисања. Дате су опште анализе и поређења улазних програма који су обрађени у дисертацији. Сам процес трансформација није специфично везан за улазе добијене из представљених преводилаца. Програм `hill_climbing` је могуће применити на разнолике улазне програме, при чему се наравно не могу увек очекивати исти нивои побољшања нивоа апстракције.

## VI СПИСАК НАУЧНИХ И СТРУЧНИХ РАДОВА КОЈИ СУ ОБЈАВЉЕНИ ИЛИ ПРИХВАЋЕНИ ЗА ОБЈАВЉИВАЊЕ НА ОСНОВУ РЕЗУЛТАТА ИСТРАЖИВАЊА У ОКВИРУ РАДА НА ДОКТОРСКОЈ ДИСЕРТАЦИЈИ

Од 14 радова које је кандидат до сада публикувао, 5 се директно односи на истраживања у оквиру рада на докторској дисертацији.

#### **M22 - Рад у истакнутом међународном часопису**

1. **Doni Pracner** and Zoran Budimac. "Enabling Code Transformations with FermaT on Simplified Bytecode". *Journal of Software: Evolution and Process*, Vol. 29, No. 5, e1857 (2017). doi: 10.1002/smr.1857.

#### **M33 - Саопштење са међународног скупа штампано у целини**

1. **Doni Pracner** and Zoran Budimac. "A Practical Tutorial for FermaT and WSL Transformations". In: *Proceedings of the 6th Workshop on Software Quality Analysis, Monitoring, Improvement, and Applications*. Ed. by Zoran Budimac. Vol. 1938. CEUR-WS.org. Sept. 2017, 11:01–11:08. url: <http://ceur-ws.org/Vol-1938/>
2. **Doni Pracner** and Zoran Budimac. "Transforming Low-level Languages Using FermaT and WSL". In: *Proceedings of the 2nd Workshop on Software Quality Analysis, Monitoring, Improvement, and Applications*. Ed. by Zoran Budimac. Vol. 1053. CEUR-WS.org. Sept. 2013, pp. 71–78. url: <http://ceur-ws.org/Vol-1053/>.
3. **Doni Pracner** and Zoran Budimac. "Restructuring Assembly Code Using Formal Transformations". In: *Proc. of Symposium on Computer Languages, Implementations and Tools (SCLIT 2011) held within Int. Conf. on Numerical Analysis and Applied Mathematics ICNAAM* Ed. by Theodore E. Simos. Vol. 1389. AIP proceedings. Kassandra, Halkidiki, Greece, Sept. 2011, pp. 845–848. isbn: 978-0-7354-0956-9. doi: <http://dx.doi.org/10.1063/1.3636865>. Url: <http://scitation.aip.org/content/aip/proceeding/aipcp/1389>.
4. **Doni Pracner** and Zoran Budimac. "Understanding Old Assembly Code Using WSL". In: *Proc. of the 14th International Multiconference on Information Society (IS 2011)*. Ed. by Marko Bohanec et al. Vol. A. Ljubljana, Slovenia: Jožef Stefan Institute, Ljubljana, Oct. 2011, pp. 171–174. isbn: 978-961-264-035-4. url: <http://is.ijs.si>.

## **VII ЗАКЉУЧЦИ ОДНОСНО РЕЗУЛТАТИ ИСТРАЖИВАЊА**

Кандидат у оквиру дисертације представља приступ раду са кодом ниског нивоа који омогућава аутоматско реструктурирање и подизање на виши ниво апстракције. Приступ је описан као процес који се састоји из више фаза, а ослања се на постојећи систем за трансформације FermaT и језик WSL. У оквиру тог система постоји велик број трансформација које очувавају семантику програма и као такве су јако погодне за реструктурирање кода.

Представљени процес је дизајниран тако да буде флексибилан и фазе у њему су независне. Ово значи да је лако прилагодити процес тренутним потребама, али и употребити развијене алате за друге процесе. Процес се типично састоји од два дела: превођење програма ниског нивоа у поменути језик; и трансформисање тих превода.

У оквиру дисертације су представљена два алата за превођење. Први ради са подскупом x86 асемблера, док је други предвиђен за програме у Микро Јава виртуелној машини. Резултати ових превода су програми ниског нивоа у језику WSL.

Трансформације се могу ручно бирати и примењивати, међутим то захтева искусног корисника система. У оквиру дисертације је представљен програм који омогућава аутоматски одабир и примену трансформација, што га чини прикладним и за кориснике без икаквог искуства са системом. Главни поступак аутоматизације се заснива на алгоритму претраживања успоном (eng. *hill climbing*). Принципијелно се састоји од примена трансформација из унапред одабраног скупа на произвољна места у програму. Новодобијени програм се тада пореди са тренутном верзијом у складу са неком функцијом подобности (eng. *fitness function*). Уколико је нови програм бољи, узима се за основу за даље трансформације, а ако није, једноставно се одбацује. Поступак се зауставља када ниједна трансформација на тренутном програму не доводи до побољшања. У дисертацији је представљен поступак који се ослања на структурну метрику (eng. *structure*) као функцију подобности. Ова метрика је уграђена у систем FermaT и у основи даје тежинску суму структуралних елемената програма. У оквиру ове метрике, на пример, безусловни скокови имају велике тежине, док конструкције високог нивоа, као што су петље са јасним условом имају мање тежине, а најмање тежине имају обичне наредбе.

У дисертацији су представљени резултати експеримената над различитим типовима улазних

програма. Прву групу представљају програми преведени из асемблера алатом `asm2ws1`. Другу главну групу представљају програми преведени из Микро Јава бајткода алатом `mjc2ws1`. У оквиру другог алата постоји и неколико прекидача који утичу на добијене преводе. Могуће је променити начин рада са стеком, представљање привремених променљивих, као и начин складиштења локалних променљивих у процедурама. У дисертацији је представљено осам варијанти превода истих програма који су сви реструктурирани истим аутоматским трансформатором. Код свих врста улаза су добијена значајна унапређења структурираности програма по свим расположивим метрикама, али се проценти ипак разликују по групама. Добијени резултати су искоришћени за анализу самог процеса, а најповољнија варијанта је коришћена за даље експерименте. Због природе процеса који непрестано примењује трансформације на све делове програма, ранији напреси који скраћују програм значајно скраћују и сам процес. Самим тим су резултати са највећим унапређењима метрика добијени у најкраћем времену, док код неких од мало слабијих резултата постоје повећања трајања извршавања и за неколико редова величина. Урађене су анализе утицаја метрика улазних програма на време извршавања трансформација и нису нађене стриктне корелације повећања времена. За програме са сличним метрикама процес може да има значајно различита времена трајања, најдужа забележена времена нису са програмима са највећим метрикама, нити су најкраћа времена код програма са најмањим метрикама. Раст у времену у општем случају се очекује код већих програма, само није јасно колико веће време се може очекивати, већ је значајније колико брзо трансформатор може да нађе начине да реструктурира и скрати неке делове улазног програма.

Процес значајно зависи од одабране функције погодности. У складу са теоремом да „бесплатни ручак не постоји“ у проблемима претраживања и оптимизације, у општем случају је немогуће направити функцију погодности тако да доводи до најбољих резултата код произвољних улазних програма. Са друге стране, програми за које је предвиђен процес имају одређене карактеристике које се могу узети у обзир при разматрању функције погодности. Примарно су у питању програми ниског нивоа, који имају велик број безусловних скокова, а са друге стране ретко или никад немају у себи структуре карактеристичне за програме високог нивоа. Самим тим, одабрана структурална метрика на иоле структурираној еквивалентној верзији улазног програма треба да да ниже вредности, односно да потврди да је та верзија погоднија и да води процес трансформације у добром правцу. На основу тога се може узети да је одабрана функција у просеку погодна за примену у овом домену улазних програма. Отворено је питање да ли постоје функције које су још боље, нарочито уколико се посматрају различити домени примена.

Природа процеса је таква да се сигурно завршава у коначном времену без обзира на особине произвољног улазног програма. Наиме, нове верзије програма за унапређивање се бирају искључиво ако имају нижу вредност метрике, те није могуће да ће доћи до уласка у бесконачну петљу између две или више верзија. На било коју појединачну верзију програма је могуће применити коначно много трансформација, а уколико се не нађе унапређење, поступак се прекида. У дисертацији је дат пример у коме би се са најмање четири додатне трансформације програм могао додатно упростити, односно довести до стања у коме је тражена метрика нижа. У општем случају је непознато за произвољни улазни програм колико корака унапред би се морало гледати да би се нашла боља верзија, односно чак није познато ни да ли постоји боља верзија до које би се могло стићи у коначно много корака, пошто је ово варијација тзв. халтинг проблема.

## VIII ОЦЕНА НАЧИНА ПРИКАЗА И ТУМАЧЕЊА РЕЗУЛТАТА ИСТРАЖИВАЊА

Комисија позитивно оцењује начин приказа и тумачења резултата. Оригинални резултати дисертације приказани су систематично и јасно, уз употребу садржајних табела и графикана, тумаче се коректно, и дискутују у склопу претходних релевантних истраживања.

## IX КОНАЧНА ОЦЕНА ДОКТОРСКЕ ДИСЕРТАЦИЈЕ:

1. Да ли је дисертација написана у складу са образложењем наведеним у пријави теме

Увидом у образложење наведено у пријави теме може се утврдити да је дисертација написана у складу са планом истраживања наведеним у пријави теме и да добијени резултати одговарају циљевима постављеним у пријави теме.

2. Да ли дисертација садржи све битне елементе

Дисертација садржи све битне елементе научно-истраживачког рада: неопходну теоријску основу, приказ релевантних истраживања из области, и оригиналне методолошке и емпиријске доприносе који унапређују постојећа знања и схватања у домену превођења и трансформисања програма ниског нивоа. Списак референци садржи релевантне радове и сведочи да кандидат одлично познаје област истраживања. Дисертација је прегледна и добро организована.

Докторка дисертација је у библиотеци Природно-математичког факултета проверена софтвером за детекцију плагијаризма iThenticate, који је показао да је индекс сличности 15%. Највеће сличности су 2-4% и односе се на техничке делове који се понављају из корисничких упутстава, што је и наглашено у раду. Остале сличности су махом са радовима кандидата.

3. По чему је дисертација оригиналан допринос науци

У дисертацији је представљен поступак за аутоматски одабир и примену трансформација ради подизања нивоа апстракције програма, у смислу трансформисања структуре програма тако да се добије програм еквивалентне функционалности написан на начин својственији програмским језицима вишег нивоа, насупрот машинским/асемблерским језицима. Поступак се заснива на алгоритму претраживања успоном, а користи се функција подобности за поређење новодобијених верзија и одлуку да ли се даље наставља са њима. Према најбољим сазнањима аутора дисертације, као и комисије, алгоритам претраживања успоном до сада није примењиван на овај тип проблема.

Представљено је више експеримената аутоматске трансформације са различитим улазним програмима добијеним из различитих преводиоца и са различитим опцијама. Улазе карактеришу различити типови система акција (рекурзивни или регуларни, који имају само скокове), као и различити начини обрађивања истих операција, као што је приступ стеку. Анализирана је успешност појединачних експеримената, али су сви резултати заједно коришћени и за боље разумевање самог процеса и типова програма који су погоднији за тренутну верзију трансформатора.

Допринос дисертације представљају и нови алати за превођење из језика ниског нивоа у језик WSL: `asm2wsl` и `mjc2wsl`, оба објављена под лиценцама отвореног кода. Први преводи подкуп x86 асемблера, а други Мирко Јава бајткод. Аллати имају потенцијал да подстакну даља (независна) истраживања у области.

Систем за трансформације `FermaT` је унапређен у сарадњи са аутором система, др Мартином Вордом. Додато је неколико нових трансформација, а неке од постојећих су унапређене. Додато је препознавање нумеричких кодова за знакове у оквиру поједностављивача израза. Додат је и нови „тихи“ мод рада, са мање порука о стању система. Додат је и нов начин прослеђивања параметара из командне линије систему `FermaT`. Посматрани заједно, описана унапређења могу подстаћи даља истраживања у области дисертације, а и шире.

4. Недостаци дисертације и њихов утицај на резултат истраживања

Није уочено да дисертација има недостатака.

**X ПРЕДЛОГ:**

На основу укупне оцене дисертације, комисија предлаже: да се докторска дисертација под називом **„Translation and Transformation of Low Level Programs“ (Превођење и трансформисање програма ниског нивоа)** кандидата **Донија Працнера** прихвати, а кандидату одобри јавна одбрана дисертације.

ПОТПИСИ ЧЛАНОВА КОМИСИЈЕ

---

др Милош Радовановић, ванредни професор, председник

---

др Зоран Будимац, редовни професор, ментор

---

др Гордана Ракић, доцент, члан

---

др Захарије Радивојевић, ванредни професор, члан