



ЗЕЛЕНЕ ПРОГРАММНЕ ОБЕСПЕЧЕННЯ

Практикум

GREEN SOFTWARE



2015



ЗЕЛЕНЕ ПРОГРАММНЕ ОБЕСПЕЧЕННЯ

GREEN SOFTWARE

• FOUNDATION OF GREEN SOFTWARE (SW)

Models, metrics and tools for research of SW power consumption



• RELIABILITY AND STABILITY ASSESSMENT OF GREEN SW

Models, techniques and tools "SRATS" and "SOTIS"



• METHODS OF DEVELOPING GREEN SW

The impact of code optimization, languages and compilers to SW energy



• DEVELOPING ECOLOGICAL HUMAN-MACHINE INTERFACE (HMI)

Assessment of HMI safety, tool and techniques of designing ecological HMI



University of Ioannina



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Министерство образования и науки Украины
Национальный аэрокосмический университет
им. Н.Е. Жуковского «Харьковский Авиационный Институт»**

**Д. А. Маевский, Е. Ю. Маевская, А. А. Орехова,
А. Ю. Кривцов, В. С. Харченко**

Зеленое программное обеспечение

Green Software

Практикум

Под редакцией В.С. Харченко и Д. А. Маевского

Проект

530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR

Green Computing & Communication

2015

УДК 004.9+681.5
3-48

Викладено матеріали практичної частини навчального курсу «Зелене програмне забезпечення (Green Software)», підготовленого для магістрантів в рамках проекту TEMPUS «Green Computing & Communication» (530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR).

Курс присвячений методологічним та практичним питанням створення «зеленого» програмного забезпечення. Метою курсу є оволодіння знаннями про сучасний стан зелених інформаційних технологій та забезпечення передумов для практичного зменшення споживання енергії комп'ютерними пристроями. Наводиться навчальна програма курсу, дається опис лабораторних робіт, методичні рекомендації з самостійного вивчення матеріалу курсу.

Для магістрантів і аспірантів університетів, які навчаються за напрямками комп'ютерних наук, комп'ютерної та програмної інженерії, а також викладачів відповідних спеціальностей.

Рецензенти: **Мищенко Віктор Олегович**, доктор технических наук, профессор, профессор кафедры моделирования систем и технологий Харьковского Национального университета имени В. Н. Каразина;
Kor Ah-Lian, Dr, Senior Lecturer, School of Computing, Creative Technologies and Engineering, Leeds Beckett University, Великобритания

Зеленое программное обеспечение. Практикум / Д. А. Маевский, Е. Ю. Маевская, А. А. Орехова, А. Ю. Кривцов, В. С. Харченко. - Под ред. Харченко В.С. и Маевского Д.А. – Харьков: Национальный аэрокосмический университет имени Н.Е. Жуковского «ХАИ». – 2015. – 161 с.

ISBN 978-966-662-716-5

3-48 Изложены материалы практической части учебного курса «Зеленое программное обеспечение (Green Software)», подготовленного для магистрантов в рамках проекта TEMPUS «Green Computing & Communication» (530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR).

Курс посвящен методологическим и практическим вопросам создания «зеленого» программного обеспечения. Целью курса является овладение знаниями о современном состоянии зеленых информационных технологий и обеспечении предпосылок для практического уменьшения потребления энергии компьютерными устройствами. Приводится учебная программа курса, дается описание лабораторных работ, методические рекомендации по самостоятельному изучению материала курса.

Для магистрантов и аспирантов университетов, обучающихся по направлениям компьютерных наук, компьютерной и программной инженерии, а также преподавателей соответствующих специальностей.

Библ. – 99 наименований, рисунков – 41, таблиц – 10

Рекомендовано к изданию Ученым советом Национального аэрокосмического университета им. Н.Е. Жуковского «Харьковский авиационный институт» (протокол № 1 от 2 сентября 2015 года).

УДК 004.9+681.5

ISBN 978-966-662-716-5

© Маевский Д.А., Маевская Е.Ю., Орехова А.А., Кривцов А.Ю., Харченко В.С.
© Национальный аэрокосмический университет имени Н.Е. Жуковского «ХАИ», 2015

СПИСОК СОКРАЩЕНИЙ

ПО – программное обеспечение

ОС – операционная система

НПП – неоднородный пуассоновский процесс

МНПО – модель надежности программного обеспечения

ДПС – динамика программных систем

ИУС – информационно-управляющая система

ЧМИ – человеко-машинный интерфейс

ПТК – программно-технический комплекс

SS – Software System

CAE – Computer-aided engineering

OS – Operation System

NHPP – Non Homogenous Poisson Process

SRGM – Software Reliability Growth Model

ВВЕДЕНИЕ

В пособии изложены материалы практической части учебного курса «Зеленое программное обеспечение» (Green Software), подготовленного для магистрантов в рамках проекта TEMPUS «Green Computing & Communication» (530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR). Курс посвящен методологическим и практическим вопросам создания «зеленого» программного обеспечения. Целью курса является овладение знаниями о современном состоянии зеленых информационных технологий и обеспечении предпосылок для практического уменьшения потребления энергии компьютерными устройствами.

Пособие содержит описания лабораторных работ и семинаров, в приложениях приведены учебная программа курса и методические рекомендации по самостоятельному изучению материалов курса.

В первый раздел включены лабораторные работы по модулю «Модели и метрики зеленого программного обеспечения». Данные лабораторные работы направлены на изучение методов сравнительной оценки степени «зелёности» программ, а также методике измерения абсолютного уровня электроэнергии, потребляемой персональным компьютером.

Второй раздел посвящен важнейшему свойству всех «зеленых» программ – их надежности. В разделе описаны лабораторные работы, в ходе выполнения которых магистранты научатся использовать современные программные системы для оценки параметров надежности программного обеспечения на основе данных о динамике выявления дефектов.

Третий раздел посвящен методологии разработки зеленого программного обеспечения.

Четвертый раздел посвящен вопросам разработки экологических человеко-машинных интерфейсов информационно-управляющих систем. В разделе описаны лабораторные работы и практические занятия, которые направлены на изучение новых поколений ЧМИ, методов оценки такого свойства экологических ЧМИ, как безопасность, а также методике выбора наиболее экологического прототипа ЧМИ.

Рисунки, таблицы и формулы для удобства нумеруются в пределах каждой лабораторной работы.

Книга предназначена для магистрантов и аспирантов университетов, обучающихся по направлениям компьютерных наук, компьютерной и программной инженерии, при изучении методов и средств разработки «зеленых», энергоэффективных систем управления сложными технологическими и техническими объектами, а также может быть полезна для преподавателей, ведущих занятия по соответствующим курсам.

Пособие подготовлено заведующим кафедрой теоретических основ и общей электротехники Одесского национального политехнического университета д.т.н., профессором Маевским Д. А. (раздел 1), к.т.н., доцентом Маевской Е. Ю. (раздел 2), старшим преподавателем Донбасского государственного технического университета Кривцовым А. Ю. (раздел 3), д.т.н., профессором Харченко В. С., к.т.н., старшим преподавателем Ореховой А. А. (раздел 4). Введение и Приложение А подготовлены В. С. Харченко, Д. А. Маевским и А. Ю. Кривцовым, Приложение Б – Е. Ю. Маевской. Общее редактирование проведено В. С. Харченко и Д. А. Маевским.

АНОТАЦІЯ

УДК 004:504(045)
П22

Зелене програмне забезпечення. Практикум. /Д.А. Маєвський, О.Ю. Маєвська, А.А. Орехова, А.Ю. Кривцов, В.С. Харченко. – За ред. Харченка В.С. і Маєвського Д. А. – Харків: Національний аерокосмічний університет імені М.Є. Жуковського «ХАІ». – 2015. – 161 с.

Викладено матеріали практичної частини навчального курсу «Зелене програмне забезпечення (Green Software)», підготовленого для магістрантів в рамках проекту TEMPUS «Green Computing & Communication» (530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR).

Курс присвячений методологічним та практичним питанням створення «зеленого» програмного забезпечення. Метою курсу є оволодіння знаннями про сучасний стан зелених інформаційних технологій та забезпечення передумов для практичного зменшення споживання енергії комп'ютерними пристроями.

Практична частина курсу складається з восьми лабораторних робіт та семінарського заняття. Практикум спрямовано на вироблення у студентів навичок створення такого програмного забезпечення, при виконанні якого рівень споживання електричної енергії обчислювальним пристроєм є мінімальним. Наводиться навчальна програма курсу, перелік запитань та літературних джерел для самостійного вивчення матеріалу курсу.

Для магістрантів і аспірантів університетів, які навчаються за напрямками комп'ютерних наук, комп'ютерної та програмної інженерії, при вивченні методів і засобів програмної інженерії, а також для викладачів дисциплін за відповідними напрямками.

Бібл. – 99 найменувань, рисунків – 41, таблиць – 10

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	3
ВСТУП.....	4
1. МОДЕЛІ ТА МЕТРИКИ ЗЕЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	6
1.1. Лабораторна робота №1. Використання метричних оцінок для порівняння програмного забезпечення по ступеню «зеленості»	6
1.2. Лабораторна робота № 2. Методи і засоби експериментальних досліджень енергоспоживання персонального комп'ютера.....	16
2. МЕТОДИ ОЦІНКИ НАДІЙНОСТІ Й СТІЙКОСТІ ЗЕЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	23
2.1. Лабораторна робота №3. Моделювання надійності в програмній системі SRATS	23
2.2. Лабораторна робота №4. Моделювання надійності в програмній системі SOTIS.....	37
3. МЕТОДИ РОЗРОБКИ ЗЕЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	60
3.1. Лабораторна робота № 5. Вплив оптимізації програмного коду на енергоспоживання програмного забезпечення	60
3.2. Лабораторна робота № 6. Вплив мови програмування та компілятора на енергоспоживання програмного забезпечення	70
4. РОЗРОБКА ЕКОЛОГІЧНИХ ЛЮДИНО-МАШИННИХ ІНТЕРФЕЙСІВ	79
4.1. Лабораторна робота № 7. Оцінка функціональної безпеки екологічних людино-машинних інтерфейсів на основі ризик-аналізу.....	79
4.2. Лабораторна робота № 8. Вибір прототипу людино-машинного інтерфейсу на основі моделі якості (екологічності).....	98
4.3. Семінар. Аналіз методів проектування екологічних людино-машинних інтерфейсів	117
ЛІТЕРАТУРА.....	123
АНОТАЦІЯ ТА ЗМІСТ	133
ABSTRACT AND CONTENT	136
ДОДАТОК А. НАВЧАЛЬНА ПРОГРАММА	138
ДОДАТОК Б. МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ПО САМОСТІЙНІЙ РОБОТІ.....	151

ABSTRACT

UDC 004:504(045)

Green Software. Practical workshop. / D. A. Maevsky, E. J. Maevskaja, A. A. Orechova, A. Y. Kryvtsov, V. S. Kharchenko. - Ed. Kharchenko V. S. and Maevsky D. A. – Kharkiv: National Aerospace University «KhAI». – 2015. – 161 c.

ISBN

The material practical part of the course "Green Software», for undergraduates trained under the project TEMPUS «Green Computing & Communication» (530270-TEMPUS-1-2012-1-UK-TEMPUS-JPCR).

The course is devoted to methodological and practical problems of creating "green" software. The aim of the course is mastery of knowledge about the current state of green IT and software prerequisites for practical reduction of energy consumption of computer devices.

The practical part of the course consists of eight laboratory work and one seminars. The workshop aims to develop students' skills of creating such software, the performance of which the level of electricity consumption is minimal computing device. An educational program of the course, a list of questions and literature for self-study course material.

For university undergraduates and graduate students who study in areas of computer science, computer and software engineering, the study of methods and tools for software engineering, as well as for teachers in chosen disciplines.

Ref. – 99 items, figures – 41, tables – 10

CONTENT

ABBREVIATION 3

INTRODACTION 4

1. MODELS AND METRICS OF GREEN SOFTWARE6

1.1. Lab1. Using metric estimates to compare the software on the degree of "greenness" 6

1.2. Lab 2. Methods and means of experimental studies power consumption of the personal computer..... 16

2. RELIABILITY AND STABILITY ASSESSMENT METHODS FOR GREEN SOFTWARE 23

2.1. Lab 3. Simulation of software reliability in instrumental tool “SRATS” 23

2.2. Lab 4. Simulation of software reliability in instrumental tool “SOTIS” 37

3. METHODS OF DEVELOPING GREEN SOFTWARE60

3.1. Lab 5. The impact of code optimization to software energy 60

3.2. Lab 6. Impact of programming languages and compiler to software energy 70

4. DEVELOPMENT OF ECOLOGICAL HUMAN-MACHINE INTERFACE 79

4.1. Lab 7. Assessment of functional safety environmental human-machine interfaces based on risk analysis 79

4.2. Lab 8. Choice prototype of human-machine interface based on the model of quality (ecological)..... 98

4.3. Seminar. Analysis of methods for designing ecological human-machine interfaces 117

REFERENCES 123

ABSTRACT 136

CONTENT 137

APPENDIX A. COURSE PROGRAM 138

APPENDIX B. TUTORIAL RECOMMENDATION 151

ПРИЛОЖЕНИЕ А. УЧЕБНАЯ ПРОГРАММА

DESCRIPTION OF THE MODULE

TITLE OF THE MODULE	Code
Green Software	PhD7

Teacher(s)	Department
<p>Coordinating: Prof. D. A. Maevsky, Prof V. S. Kharchenko</p> <p>Others: Dr. E. J. Maevskaya, PhD students E. D. Maevskaya, A. Yu. Kryvtsov, Dr A. A. Orekhova</p>	Computer intellectual systems and networks

Study cycle	Level of the module	Type of the module
PhD	B	Full-time tuition

Form of delivery	Duration	Language(s)
Full-time tuition	1 sem	Russian / English

Prerequisites	
<p>Prerequisites: Mathematic analysis, basic knowledge of modern computer systems and information technologies, principles and methods of system analysis</p>	<p>Co-requisites (if necessary):</p>

Credits of the module	Total student workload	Contact hours	Individual work hours
3	90	46	44

Aim of the module (course unit): competences foreseen by the study program
<p>The aim of module is to create a knowledge base on modern state of Green Information Technologies and to provide prerequisites for practical decreasing of energy consumption by computer devices by way of creating the Green Software. Application of acquired knowledge: system programming, programming mobile applications, energy management The study also expands the current research on software engineering by combining green technology and holistic approach to creating green software in the context of all software life cycle.</p>

Learning outcomes of module (course unit)	Teaching/learning methods	Assessment methods
At the end of course, the successful student will be able to: 1. Perform assessment of greenness software	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
2. Select the best software on the basis of electric power that is consumed during its execution	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
3. Perform an assessment of the absolute level of consumption of electric energy	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
4. To carry out the measurement of electrical energy that is consumed computer	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
5. Choose a software environment for the most effective implementation of the developed software	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire
6. Apply the basic principles of effective green programming	Interactive lectures, Learning in laboratories, Just-in-Time Teaching	Module Evaluation Questionnaire

Themes	Contact work hours						Time and tasks for individual work		
	Lectures	Consultations	Seminars	Practical work	Laboratory work	Placements	Total contact work	Individual work	Tasks
MP7.1 Models and metrics of green software									
1. Introduction in Green Software 1.1. The purpose and tasks of the course. 1.2. What is Software? 1.3. Software as an artificial system. 1.4. The main functions of Software. 1.5. Two directions of Software influence on power and resources	2						2	2	1.6. How can Software become “green”?
2. Influence of software on internal and external computing resources 2.1. The concept of internal (system) resources. 2.2. Software influence on system resource consumption. 2.3. Algorithm optimization as opposition means to bloat effect	2			2			4	2	2.4. The effect of “program bloat” and its negative consequences
3. “Shades of green” – system of software reliability metrics. 3.1. The need to use metrics 3.2. Types of metrics 3.3. Metrics code	2			2			4	2	3.4. Examples of the use of metrics
4. Unresolved problems of Green Software and the prospects of the researches 4.1. The need for standardization of Green Software. 4.2. The problems of quality	2						2	4	4/5. The need for Green Programming method development. 4.6.The

standard development. 4.3. The problem accurate reliability estimation and the ways of its solution. 4.4. The problem of Software attribution.									problem of the external attack on Software opposition and the ways of its solution
MP7.2 Methods of reliable and sustainable software									
5. Traditional approach to the Problem. Software Reliability instrumental tools overview. 5.1. The concept of Software reliability. 5.2. Defects as the factor of reliability decreases. 5.3. The concept of the secondary defects and their accounting problem. 5.4. The necessity of accuracy increases in estimating the reliability. 5.5. Software Reliability instrumental Tools	2						2	4	5.6. Modern methods and technologies of the reliability simulation and estimation
6. Software System Dynamics as a new approach to the Software reliability. 6.1. Basic concepts of the theory of the dynamics of software systems. 6.2. The concept of a flow of primary and secondary defects. 6.3. The basic equations of the dynamics of software systems and solutions. 6.4. Modeling of flow defects	2				2		4	4	
7. Sustainable software from the point of view of Software System Dynamics. 7.1. The concept of Software stability. 7.2. The types of external impacts and their consequences.	2				2		4	4	7.3. The ways of external impact parrying

MM7.3 Methods of development green software									
8. Development cycle of green software 8.1 Development stages 8.2 Requirements analysis 8.3 Design of green software	2						2	4	8.4 Model of software life cycle
9 Building of green software 9.1 Algorithmic optimization 9.2 Source code optimization 9.3 Idle efficiency 9.4 Context awareness	2			2			4	4	9.5 Effectiveness of data structures
10. Compile of green software 10.1 Optimization in compilers 10.2 Green techniques for compilers 10.3 Green compilers	2			2			4	4	10.4 Compiling interpreted languages
11. Testing of green software 11.1 Compiling and coding efficiency testing 11.2 Design efficiency testing 11.3 Requirements analysis efficiency testing	2						2	4	11.4 Double and triple V-model for testing and verification
MM7.4 Development of ecological human-machine interface									
12. Green human-machine interfaces of critical information and control systems 12.1 The basic principles of designing HMI 12.2 Quality model and basic properties of green HMI 12.3 Metrics for assessment green HMI 12.4 The process of metric assessment HMI 12.5 Tools for assessing green HMI	2		2		2		6	4	12.6 Quality model of HMI

13 Methods of safety assessment green human-machine interfaces 13.1 Methodology for safety assessing critical systems 13.2 Methods of risk analysis. 13.3 Choosing and adapting risk analysis methods for safety assessment HMI 13.4 Combination of methods 13.5 The tools of risk analysis	2		2		2		6	4	13.6 Choosing the prototype of HMI
Total	26		4		16		46	44	90

Assessment strategy	Weight in %	Deadlines	Assessment criteria
Lecture activity, including fulfilling special self-tasks	10	7,14	<p>85% – 100% Outstanding work, showing a full grasp of all the questions answered.</p> <p>70% – 84% Perfect or near perfect answers to a high proportion of the questions answered. There should be a thorough understanding and appreciation of the material.</p> <p>60% – 69% A very good knowledge of much of the important material, possibly excellent in places, but with a limited account of some significant topics.</p> <p>50% – 59% There should be a good grasp of several important topics, but with only a limited understanding or ability in places. There may be significant omissions.</p> <p>45% – 49% Students will show some relevant knowledge of some of the issues involved, but with a good grasp of only a minority of the material. Some topics may be answered well, but others will be either omitted or incorrect.</p> <p>40% – 44% There should be some work of some merit. There may be a few topics answered partly or there may be scattered or perfunctory</p>

			<p>knowledge across a larger range.</p> <p>20% – 39% There should be substantial deficiencies, or no answers, across large parts of the topics set, but with a little relevant and correct material in places.</p> <p>0% – 19% Very little or nothing that is correct and relevant.</p>
Learning in laboratories	30	7,14	<p>85% – 100% An outstanding piece of work, superbly organised and presented, excellent achievement of the objectives, evidence of original thought.</p> <p>70% – 84% Students will show a thorough understanding and appreciation of the material, producing work without significant error or omission. Objectives achieved well. Excellent organization and presentation.</p> <p>60% – 69% Students will show a clear understanding of the issues involved and the work should be well written and well organized. Good work towards the objectives.</p> <p>The exercise should show evidence that the student has thought about the topic and has not simply reproduced standard solutions or arguments.</p> <p>50% – 59% The work should show evidence that the student has a reasonable understanding of the basic material. There may be some signs of weakness, but overall the grasp of the topic should be sound. The presentation and organization should be reasonably clear, and the objectives should at least be partially achieved.</p> <p>45% – 49% Students will show some appreciation of the issues involved. The exercise will indicate a basic understanding of the topic, but will not have gone beyond this, and there may well be signs of confusion about more complex material. There should be fair</p>

			<p>work towards the laboratory work objectives.</p> <p>40% – 44% There should be some work towards the laboratory work objectives, but significant issues are likely to be neglected, and there will be little or no appreciation of the complexity of the problem.</p> <p>20% – 39% The work may contain some correct and relevant material, but most issues are neglected or are covered incorrectly. There should be some signs of appreciation of the laboratory work requirements.</p> <p>0% – 19% Very little or nothing that is correct and relevant and no real appreciation of the laboratory work requirements.</p>
Module Evaluation Quest	60	8,16	The score corresponds to the percentage of correct answers to the test questions

Author	Year of issue	Title	No of periodical or volume	Place of printing. Printing house or internet link
Compulsory literature				
MP7.1 Models and metrics of green software				
Kern E., Dick M., Naumann S., Guldner A., Johann T.	2013	Green Software and Green Software Engineering – Definitions, Measurements, and Quality Aspects		Zurich : [ETH]
	1993	ISO/IEC 2382-1:1993, Information technology – Vocabulary – Part 1: Fundamental terms		http://webstore.ansi.org/RecordDetail.aspx?sku=ISO%2FIEC%202382-1:1993&sourcekey=word=_inurl:webstore.ansi.org%23inurl:sku%3Diso&source=google&adgro

				up=iso13&gclid=C Lz- kaDjzcUCFeXVcg odpxYAog
Clarke R. A., Knake R. K	2010	Cyberwar. The Next Threat to National Security and What to Do About It		Ecco: HarperCollins
	2011	ISO/IEC 25010:2011. Systems and software Quality Requirements and Evaluation (SQuARE) – System and software quality models.		http://www. standards.ru/doc ument/4580604.a spx
Burrows S., S. Tahaghoghi	2007	Source code authorship attribution using n- grams		Melbourne, Australia. – RMIT University
MP7.2 Methods of reliable and sustainable software				
D.A. Maevsky, Igor A. Ushakov, Ludmila N. Shapa.	2011	Software reliability. What is it?	Vol.8. – № 4(31).	http://www.gneden ko- forum.org/Journal/ 2013/042013/RTA _4_2013-07.pdf
Maevsky, D. A	2013	A New Approach to Software Reliability	№ 8166	Berlin : Springer
Lyu M. R.	1996	Handbook of Software Reliability Engineering		New York: McGraw-Hill Company
H. Okamura, T. Dohi	2013	SRATS: Software reliability assessment tool on spreadsheet (Experience report)	No. 32	IEEE 24th International Symposium on Software Reliability Engineering (ISSRE)
Jatain, A, Mehta, Y.	2014	Metrics and models for Software Reliability: A	No. 19	International Conference on Issues and

		systematic review		Challenges in Intelligent Computing Techniques (ICICT)
MM7.3 Methods of development green software				
Vyacheslav Kharchenko	2014	Green IT-Engineering. In 2 Volumes		National Aerospace University KhAI, Kharkiv, Ukraine
M.F. McDonald	2002	Integrating green into an existing management system: Performing a “green” gap analysis.	Vol. 56, № 1	Annual Quality Congress
M. Dick, J. Drangmeister, E. Kern, S. Naumann	2013	Green software engineering with agile methods		2nd International Workshop on Green and Sustainable Software (GREENS 2013)
B. Steigerwald, A. Agrawal	2011	Developing Green Software		http://software.intel.com/sites/default/files/m/0/6/7/8/1/37258-developing_green_software.pdf
S. Murugesan, G. R. Gangadharan	2012	Harnessing Green IT: Principles and Practices		Wiley
D. Bicknell	2011	8 ways to make your software applications more energy efficient		http://www.computerweekly.com/blogs/greentech/
F. Fakhar, B. Javed, R. Rasool, O. Malik, K. Zulfiqar	2012	Software level green computing for large scale systems	Vol. 1, №4	Journal of Cloud Computing: Advances, Systems and Applications
J. Visser.	2012	Green Software		http://staff.science.uva.nl/~delaat/news/2012-03-23/slides_visser.pdf

MM7.4 Development of ecological human-machine interface				
Vyacheslav Kharchenko	2014	Green IT-Engineering. In 2 Volumes		National Aerospace University KhAI, Kharkiv, Ukraine
Orekhova A.	2014	Human-machine interface quality assessment techniques: Green and safety issues		Proceeding of 10th IEEE International Conference on Digital Technologies “DT 2014”
	2011	ISO/IEC 25010:2011 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE)		http://www.iso.org/iso/home/store/catalogue_tc.htm
	2010	ISO/IEC 9241-210:2010: Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems		http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075
Andrew Rae	2011	Helping the operator in the loop: practical human machine interface principles for safe computer controlled systems	Vol. 86.	Proceeding SCS '07 Proceedings of the twelfth Australian workshop on Safety critical systems and software and safety-related programmable systems

Orekhova A.	2014	The Cooperative Human-Machine Interfaces for Cloud-Based Advanced Driver Assistance Systems: Dynamic Analysis and Assurance of Vehicle Safety		Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014)
Additional literature				
MP7.1 Models and metrics of green software				
N. Rosenblum, X. Zhu, B. Miller	2011	Who wrote this code? Identifying the authors of program binaries		http://www.cosic.esat.kuleuven.be/esorics2011/slides/Session_04_Forensics_Biometrics_Software_Protection/0850_rosenblum.pdf
A. Ekelhart, S. Fenz, M. Klemen, E. Weippl	2007	Security Ontologies: Improving Quantitative Risk Analysis		https://www.sba-research.org/wp-content/uploads/publications/2007%20-%20Ekelhart%20-%20Security%20Ontologies%20Improving%20Quantitative%20Risk%20Analysis.pdf
MP7.2 Methods of reliable and sustainable software				
Jun A, Hanyu P, Liang Y.	2014	Software Reliability Virtual Testing for Reliability Assessment.	No. 19	IEEE Eighth International Conference on Software Security and Reliability-Companion (SERE-C)
Bidhan K, Awasthi A.	2014	Estimation of reliability parameters of software growth models using a variation of Particle Swarm Optimization.	No. 20	Confluence The Next Generation Information Technology Summit

MM7.3 Methods of development green software				
W. Fornaciari, P. Gubian, D. Sciuto, Silvano	1988	Power estimation of embedded systems: A hardware/software codesign approach	Vol. 6/2	IEEE Trans. on VLSI Systems.
E. Chung	2009	Software Approaches for Energy-efficient System Design: Focused on Dynamic Power Management and Program Specialization		VDM Publishing
	2011	Intel Power Checker		http://software.intel.com/en-us/blogs/2011/06/27/intel-power-checker
	2011	Intel Battery Life analyzer		https://downloadcenter.intel.com/Detail_Desc.aspx?agr=Y&DwnldID=19351
	2011	Microsoft Joulemeter		http://research.microsoft.com/en-us/projects/joulemeter/
	2011	WattsUp		https://www.wattsupmeters.com/secure/index.php
MM7.4 Development of ecological human-machine interface				
Lau N.	2008	Ecological interface design in the nuclear domain: an application to the secondary subsystems of a boiling water reactor plant simulator	V. 55, № 6.	IEEE Transactions on nuclear science

ПРИЛОЖЕНИЕ Б. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ

Пояснения к учебной программе

Самостоятельную работу над дисциплиной «Зеленое программное обеспечение» («Green Software») следует начинать с изучения учебной программы, которая приведена в данном Приложении. Эта программа включает следующие элементы.

Объект изучения – зеленые информационные технологии

Предмет изучения – предпосылки для практического уменьшения потребления энергии компьютерными устройствами.

Требования к исходным знаниям и навыкам, которые необходимо иметь перед началом изучения:

- Базовые знания в области современных компьютерных систем и информационных технологий.
- Математический анализ.
- Принципы и методы системного анализа.

Целью изучения дисциплины является получение теоретических знаний и практических умений в области создания энергоэкономичного и надежного программного обеспечения

В результате ее изучения обучаемые должны научиться:

- анализировать и синтезировать информацию;
- задавать и отвечать на поставленные вопросы, мыслить креативно и критически;
- предпринимать исследовательские действия и оценивать получаемые результаты с использованием качественных и количественных показателей;
- формулировать практические решения проблем, эффективно использовать время и доступные ресурсы для достижения целей дисциплины;
- демонстрировать гибкость, адаптируемость, самомотивацию и инициативу, умение выразить свое мнение.

- совершенствовать и развивать свой интеллектуальный и общекультурный уровень;
- проявлять способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности;
- применять перспективные методы исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий

В результате изучения дисциплины студенты должны:

1) Теоретическая составляющая:

- Выполнять оценки степени зелены программы обеспечения
- Выполнять оценки абсолютного уровня потребления электрической энергии
- Выполнять оценки экологических человеко-машинных интерфейсов

2) Познавательная составляющая:

- Ознакомление с измерениями электрической энергии, которая потребляется компьютером
- Выбор программной среды для наиболее эффективной реализации разработанного программного обеспечения
- Ознакомление с характеристиками и метриками экологических человеко-машинных интерфейсов

3) Практическая составляющая:

- Умение выбирать лучшее программное обеспечение на основе электрической мощности, которая потребляется в процессе его выполнения
- Применение основных принципов эффективного зеленого программирования
- Умение выбирать прототип человеко-машинного интерфейса на основе свойств экологичности.

– Проводить анализ безопасности экологического человеко-машинного интерфейса.

Структура и содержание модулей. Дисциплина включает три модуля:

МР7.1. Основы разработки зеленого программного обеспечения

Лекционный курс

Тема 1. Введение в зеленое программное обеспечение. Общая характеристика курса. Цель и задачи курса. Что такое программное обеспечение? Программное обеспечение как искусственная система. Основные функции программного обеспечения. Как программное обеспечение может стать "зеленым"? Два направления влияния программного обеспечения на энергию и ресурсы.

Тема 2. Влияние программного обеспечения на внутренние и внешние ресурсы вычислительных систем.

Концепция внутренних ресурсов. Программное обеспечение и влияние на потребление системных ресурсов. Алгоритм оптимизации как противодействие эффекту раздувания.

Тема 3. Метрики «оттенков зеленого» в программном обеспечении.

Необходимость использования метрик. Виды метрик. Метрики программного кода.

Тема 4. Нерешенные проблемы в зеленом программном обеспечении и пути их решения.

Необходимость стандартизации зеленого программного обеспечения. Проблемы разработки стандарта качества. Точная оценка надежности: проблема и пути ее решения. Проблема атрибуции программного обеспечения.

Лабораторные работы

Работа 1. Использование метрических оценок для сравнения степени «зелености» программ (2 часа).

Целью работы является сравнение двух или большего числа программ по критерию степени их «зелености». Степень зелености определяется на основании метрик «оттенков зеленого», которые изложены в лекции 2.

Работа 2. Экспериментальные исследования энергопотребления персонального компьютера (2 часа)

Эта лабораторная работа заключается в измерении абсолютного уровня потребления электрической энергии при выполнении различных программ. Измерение выполняется прибором РМ- 231.

МР7.2 Методы надежного и устойчивого программного обеспечения

Лекционный курс

Тема 1. Традиционный подход к проблеме надежности программного обеспечения. Обзор инструментальных средств оценки надежности.

Основные понятия надежности. Модели надежности и их эволюция. Достоинства и недостатки моделей. Общие проблемы современной теории надежности. Инструментальные средства оценки надежности.

Тема 2. Теория динамики программных систем как новый подход к надежности ПО

Основные концепции теории динамики программных систем. Понятие потоков первичных и вторичных дефектов. Основные уравнения динамики программных систем и их решение. Моделирование потоков дефектов.

Тема 3. Устойчивость программных систем в свете теории динамики.

Общие понятия устойчивости. Условия устойчивости программных систем.

Лабораторные работы

Работа 3. Моделирование надежности в программной системе SRATS (2 часа).

Изучение интерфейса и практической работы в SRATS. Подготовка и ввод исходных данных. Интерпретация результатов

Работа 4. Моделирование надежности в программной системе SOTIS (2 часа).

Изучение интерфейса и практической работы в SOTIS. Подготовка и ввод исходных данных. Интервалы в рядах исходных данных. Интервальный анализ надежности. Изучение процессов в программной системе на основе результатов интервального анализа

МР7.3 Методы разработки зеленого программного обеспечения

Лекционный курс

Тема 1. Цикл разработки зеленого программного обеспечения.

Этапы разработки зеленого программного обеспечения. Анализ требований к зеленому программному обеспечению. Проектирование зеленого программного обеспечения.

Тема 2. Построение зеленого программного обеспечения.

Алгоритмическая оптимизация. Оптимизация исходного кода. Эффективность простоя. Контекстная зависимость.

Тема 3. Компиляция зеленого программного обеспечения.

Оптимизация в компиляторах. Зеленые техники для компиляторов. Зеленые компиляторы.

Тема 4. Тестирование зеленого программного обеспечения.

Тестирование эффективности компиляции и построения. Тестирование эффективности проектирования. Тестирование эффективности анализа требований.

Лабораторные работы

Работа 5. Влияние оптимизации программного кода с применение статического анализа на энергопотребление программного обеспечения (2 часа).

Изучение методов оценки энергопотребления программного обеспечения. Изучение методов оптимизации программного кода с помощью методов статического анализа.

Работа 6. Влияние языка программирования и компилятора для реализации алгоритмов на энергопотребление программного обеспечения (2 часа).

Изучение влияние выбора языка программирования и компилятора на энергопотребление программного обеспечения.

МР7.4. Разработка экологических человеко-машинных интерфейсов

Лекционный курс

Тема 1. Экологические человеко-машинные интерфейсы критических информационно-управляющих систем

Новые поколения ЧМИ. Основные принципы проектирования ЧМИ. Понятие экологических ЧМИ. Модель качества и основные свойства экологических ЧМИ. Метрики для оценки экологических ЧМИ. Процесс метрической оценки ЧМИ. Инструментальные средства оценки экологических ЧМИ.

Тема 2. Методы оценивания безопасности экологических человеко-машинных интерфейсов

Методологии оценки безопасности критических систем. Классификация методов риск-анализа. Выбор и адаптация методов риск-анализа для оценки безопасности ЧМИ. Повышение качества экспертиза за счет комплексирования методов. Инструментальные средства риск-анализа.

Лабораторные работы

Работа 7. Оценка функциональной безопасности экологических человеко-машинных интерфейсов на основе риск-анализа (2 часа).

Изучение инструментального средства RAE и практическая работа по проведению риск-анализа человеко-машинного интерфейса программно-технического комплекса. Подготовка отчета по безопасности.

Работа 8. Выбор прототипа человеко-машинного интерфейса на основе модели экологичности (2 часа).

Изучение инструментального средства MetricCalculator и практическая работа по проведению процедуры выбора наиболее экологического прототипа ЧМИ. Подготовка профиля качества. Многокритериальный анализ прототипов ЧМИ. Составление отчета по качеству.

Методы оценки

Экзамен (100 %). По окончании курса проводится 90-минутный экзамен.

Отчетность по дисциплине включает отчеты по каждому виду практического занятия, а также экзамен, который включает типовые вопросы и задачи.

Подготовка к занятиям и экзамену

При подготовке к лабораторным занятиям следует обратить внимание на уяснение целей и задач (учебных или теоретических, практических и исследовательских) и знаний, которые нужны для их выполнения. При выполнении разработок и исследований необходимо строго руководствоваться описанием и попытаться найти ответы на вопросы, приведенные в конце каждой работы. Особое внимание следует уделить формулировке выводов по результатам исследований при оформлении отчета. При самостоятельной подготовке к лабораторным работам важно правильно спланировать как свою индивидуальную, так и коллективную работу, организовать отбор и анализ необходимой литературы, подготовку к ответам на вопросы, приведенные в каждом разделе. Следует обратить внимание на вопросы, вынесенные на самостоятельное изучение, которые приводятся в программе и уточняются преподавателем.

Перечень вопросов, выносимых на самостоятельную работу

1. Какие вы знаете определения для терминов «Программное обеспечение» и «Программа»?
2. Что является основной функцией программного обеспечения?
3. Могут ли исходный код программы и ее исполняемый код считаться разными программами?
4. Может ли программное обеспечение непосредственно влиять на энергопотребление компьютера и его периферийных устройств?
5. Назовите группы технических средств, работающих под управлением программного обеспечения.
6. Почему зеленое программное обеспечение должно быть свободно от программных дефектов?
7. Что является главным фактором функционирования программных систем критического назначения?
8. Назовите известные вам кибератаки на программное обеспечение критической инфраструктуры.
9. Что такое «Устойчивое программное обеспечение»?
10. Каким требованиям должно удовлетворять зеленое программное обеспечение?
11. Перечислите возможные направления исследований в области зеленого программного обеспечения.
12. Дайте определение понятию «Качество программного обеспечения».
13. Что такое модель качества?
14. Назовите основные характеристики качества программного обеспечения согласно стандарту ISO/IEC 25010-2011.
15. Зачем нужны метрики «оттенков зеленого»?
16. Попробуйте самостоятельно предложить набор метрик для оценки «степени зелености» программного обеспечения.
17. Что такое «эффект раздувания программ»?
18. Назовите основные направления инженерии зеленого программного обеспечения.
19. Почему зеленое программное обеспечение обязательно должно быть надежным?

20. В чем вы видите недостатки современной теории надежности программного обеспечения?

21. Как Вы думаете, является ли процесс выявления дефектов в программном обеспечении случайным процессом? Для ответа на этот вопрос вспомните основные признаки случайного процесса.

22. Назовите основные направления исследований в области надежности зеленого программного обеспечения.

23. Что такое устойчивость программного обеспечения?

24. Какие вы знаете пути достижения устойчивой работы программ?

25. Что такое атрибуция программного обеспечения?

26. На каких средствах и технологиях базируется современный человеко-машинный интерфейс?

27. Какими свойствами должен обладать экологический интерфейс пользователя?

28. Чем отличается подход к проектированию человеко-машинных интерфейсов, ориентированный на пользователя от подхода, ориентированного на человека?

29. Почему экологические ЧМИ в автомобильных информационных системах позволяют повысить безопасность движения?

30. Что такое контекстно-зависимая адаптация интерфейса и за счет чего она достигается?

31. Какие вопросы возникают при разработке ЧМИ для поддержки когнитивной деятельности оператора ИУС?

32. В чем суть методологии SafetyCase? Возможно ли ее применение для интерфейсов пользователя?

33. Что понимается под термином «иммерсивный интерфейс»?

34. В чем состоит идея когнитивной графики?

35. Модель жизненного цикла программного обеспечения.

36. Эффективность структур данных.

37. Компиляция интерпретируемых языков.

38. Двойная и тройная V-модель для тестирования и верификации.

СОДЕРЖАНИЕ

СПИСОК СОКРАЩЕНИЙ.....	3
ВВЕДЕНИЕ	4
1. МОДЕЛИ И МЕТРИКИ ЗЕЛЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	6
1.1. Лабораторная работа №1. Использование метрических оценок для сравнения программного обеспечения по степени «зелености»	6
1.2. Лабораторная работа № 2. Методы и средства экспериментальных исследований энергопотребления персонального компьютера	16
2. МЕТОДЫ ОЦЕНКИ НАДЕЖНОСТИ И УСТОЙЧИВОСТИ ЗЕЛЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	23
2.1. Лабораторная работа №3. Моделирование надежности в программной системе SRATS	23
2.2. Лабораторная работа №4. Моделирование надежности в программной системе SOTIS.....	37
3. МЕТОДЫ РАЗРАБОТКИ ЗЕЛЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	60
3.1 Лабораторная работа № 5. Влияние оптимизации программного кода на энергопотребление программного обеспечения.....	60
3.2 Лабораторная работа № 6. Влияние языка программирования и компилятора на энергопотребление программного обеспечения	70
4. РАЗРАБОТКА ЭКОЛОГИЧЕСКИХ ЧЕЛОВЕКО-МАШИННЫХ ИНТЕРФЕЙСОВ.....	79
4.1. Лабораторная работа №7. Оценка функциональной безопасности экологических человеко-машинных интерфейсов на основе риск-анализа	79

4.2. Лабораторная работа №8. Выбор прототипа человеко-машинного интерфейса на основе модели качества (экологичности)	98
4.3. Семинар. Анализ методов проектирования экологических человеко-машинных интерфейсов	117
ЛИТЕРАТУРА.....	123
АНОТАЦІЯ.....	133
ЗМІСТ.....	134
ABSTRACT.....	135
CONTENT	136
ПРИЛОЖЕНИЕ А. УЧЕБНАЯ ПРОГРАММА	137
ПРИЛОЖЕНИЕ Б. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ	150

Автори:
Маєвський Дмитро Андрійович
Маєвська Олена Юріївна
Орехова Анастасія Олександрівна
Кривцов Андрій Юрійович
Харченко Вячеслав Сергійович

Зелене програмне забезпечення Green Software

Практикум
(російською мовою)

Редактори
В.С. Харченко
Д.А. Маєвський

Комп'ютерна верстка
С.В. Яковлев, Л.Д. Харченко

Зв. план 2015

Підписаний до друку 21.12.2015

Формат 60x84 1/16. Папір офс. №2. Офс. друк.

Умов. друк. арк. 9,53. Уч.-вид. л. 9,07. Наклад 200 прим.

Замовлення 43. Ціна вільна

Національний аерокосмічний університет ім. М. Є. Жуковського
"Харківський авіаційний інститут"
61070, Харків-70, вул. Чкалова, 17
<http://www.khai.edu>

Віддруковано ФОП Лисенко І. Б.

61070, Харків-70, вул. Чкалова, 17, моторний корпус, к. 147

Свідоцтво про внесення суб'єкта видавничої справи в державний реєстр
видавців, виготовлювачів і розповсюджувачів видавничої продукції
ДК №2607 от 11.09.06 р.