

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ
Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи
Кашпур О.Ф.
«26» 03 2018 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«МАТЕМАТИЧНІ ОСНОВИ ТА ТЕХНОЛОГІЇ СТВОРЕННЯ
ПРОГРАМНИХ СИСТЕМ ТА АЛГОРИТМІВ» /
“MATHEMATICAL FOUNDATIONS AND TECHNOLOGY OF
DEVELOPMENT OF SOFTWARE SYSTEMS AND ALGORITHMS”

для здобувачів освітньо-наукового рівня «доктор філософії» /
for obtaining an educational-scientific degree: Doctor of Philosophy

галузь знань	11 «Математика та статистика» / “Mathematics and Statistics”
спеціальність	113 «Прикладна математика» / “Applied Mathematics”
освітній рівень	третій (освітньо-науковий) / third (educational-scientific level)
освітньо-наукова програма	«Прикладна математика» / “Applied Mathematics”
вид дисципліни	обов'язкова / mandatory
Форма навчання	денна /
Навчальний рік	2018/2019
Рік навчання	1
Кількість кредитів ECTS	3
Мова викладання, навчання та оцінювання	англійська
Форма заключного контролю	екзамен

Викладачі: професор Нікітченко Микола Степанович, д.ф.-м.н.

Пролонговано: на 2020/2020 н.р. «15» 04 2019 р.
на 2020/2021 н.р. «30» 03 2020 р.

КИЇВ – 2018

Розробник: **Нікітченко Микола Степанович**, д. ф.-м. н., проф., завідувач кафедри теорії та технології програмування

ЗАТВЕРДЖЕНО

Завідувач кафедри теорії та технології програмування

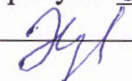


(підпис) Нікітченко М.С..

Протокол №1 від «19» 12 2017 р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «14» 02 2018 року № 6

Голова науково-методичної комісії  професор, д.ф.-м.н. Хусаїнов Д.Я.
(підпис)

1. Мета дисципліни дати сучасні знання про математичні основи та технології побудови алгоритмів та створення на їх основі програмних систем, розвинути здатності формулювати наукову проблему та робочі гіпотези на основі глибокого осмислення наявних і створення нових цілісних знань, а також професійної практики, розвивати й реалізовувати нові конкурентоздатні ідеї в галузі інформаційних технологій.

/

Discipline aim. The aim of the discipline is to provide up-to-date knowledge of mathematical foundations and technologies of algorithm construction and creation on their basis of software systems, to develop ability to formulate scientific problem and working hypotheses on the basis of deep understanding of existing and creation of new holistic knowledge, as well as professional practice, to develop and implement new competitive ideas in the field of information technology.

2. Попередні вимоги до опанування або вибору навчальної дисципліни /

Preliminary demands to master or choice of the course discipline:

1. *Знати:* основні методи побудови алгоритмів, способи їх реалізації, сучасні засоби розробки та аналізу програмних систем.
2. *Вміти:* розробляти, аналізувати та застосовувати програмні системи для розв'язання завдань та прикладних задач, використовуючи сучасні методи розробки програм.

/

1. To know: basic methods of algorithm construction, methods of algorithm implementation, modern tools for software systems development and analysis.
2. To be able to: develop, analyze and apply software systems to solve problems and applied tasks using modern software development methods.

3. Анотація навчальної дисципліни / Synopsis of the course:

Дисципліна «Математичні основи та технології створення програмних систем та алгоритмів» належить до переліку обов'язкових навчальних дисциплін. Вона забезпечує професійний розвиток аспіранта, спрямована на формування теоретичних основ та новітніх технологій розробки програмних систем, зокрема, розуміння методів їх моделювання, специфікації та верифікації на основі логіко-алгебраїчного підходу, а також вміння використовувати сучасні засоби підтримки розробки інформаційних систем, засновані на програмних логіках.

/

The discipline "Mathematical foundations and technologies of development of software systems and algorithms" belongs to the list of mandatory discipline. It provides the professional development of the PhD students, aimed at the formation of theoretical foundations and the latest technologies of software systems development, in particular, understanding of methods of their modelling, specification and verification on the basis of logic-algebraic approach, as well as ability to use modern tools of development of information systems based on program logic.

4. Завдання (навчальні цілі) / Objectives of study: набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у програмуванні, відповідно науково-освітньої кваліфікації «Доктор філософії». Зокрема, розвивати:

- здатність проведення досліджень на відповідному рівні;
- здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- здатність до побудови тлумачно-інтерпретативних схем розвинення окремих напрямків досліджень з використанням існуючих та власних теоретичних моделей і методів, що передбачає глибоке переосмислення наявних та створення нових цілісних знань;

- здатність до використання основ прикладної математики як в плані теоретичних засад постановки задач і побудови методів їх дослідження, так і в плані численних практичних застосувань і інтерпретації результатів.
- здатність оцінювати нові прикладні аспекти виконаних досліджень і виконувати їх можливу адаптацію для подальшого розвитку в прикладному напрямку.

/

Objectives (learning objectives): acquiring knowledge, skills and competences at the level of the latest achievements in programming, according to the scientific and educational qualification of “Doctor of Philosophy”. In particular, to develop:

- the ability to conduct research at an appropriate level;
- ability to search, process and analyze information from various sources;
- the ability to construct explicative and interpretative schemes for the development of particular areas of research using existing and own theoretical models and methods, which implies a deep rethinking of existing and creation of new holistic knowledge;
- ability to use the foundations of applied mathematics both in terms of theoretical principles of problem formulation and construction of methods of their study, and in terms of numerous practical applications and interpretation of results.
- the ability to evaluate new applied aspects of the investigation performed and to adapt them to further developments in the applied field.

5. Результати навчання за дисципліною / Results of learning:

Результат навчання (РН) (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН 1.1	Знати основні методи формалізації алгоритмів та програмних систем (теорія обчислюваності, програмні алгебри, мови специфікацій, програмні логіки). / To know the basic methods of construction of algorithms and software systems (computability theory, program algebras, specification languages, program logics).	<i>Лекція/ Lectures</i>	<i>Контрольна робота 1 (60% правильних відповідей), екзамен, активна робота на лекції, усні відповіді/ Test 1 (60% correct answers), exam, activity during lectures, oral answers</i>	20%
РН 1.2	Знати основні структури даних, функції та композиції, які використовуються в мовах розробки програмних систем / To know the basic data structures, functions, and compositions that are used in software development languages			
РН 1.3	Знати методи послідовного уточнення програм / To know the methods of sequential refinement of programs			20%
РН 1.4	Знати методи доведення правильності побудови програм / To know the methods of proving the correctness of program construction			
РН 2.1	Вміти застосовувати методи формалізації предметної області / To be able to apply domain formalization methods	<i>Лекція, самостійна робота/ Lectures, Individual</i>	<i>Контрольна робота 2 (60% правильних відповідей), екзамен, захист</i>	20%
РН 2.2	Вміти аналізувати функціональні вимоги до			20%

	програмних систем / To be able to analyze functional requirements for software systems	<i>work</i>	<i>проекту, виконання завдань, винесених на самостійну роботу/</i> Test 2 (60% correct answers), exam, project defense, accomplishment of tasks assigned to <i>Individual work</i>	
PH 2.3	Вміти застосовувати програмні засоби розробки систем / To be able to use systems development tools			5%
PH3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти / To be able to justify own view of the problem, communicate with colleagues in the design and development of programs, prepare written reports			5%
PH4.1	Демонстрація авторитетності, інноваційність, високий ступінь самостійності, академічна та професійна доброчесність, послідовна відданість розвитку нових ідей або процесів у передових контекстах професійної та наукової діяльності. / Demonstration of authority, innovativeness, high degree of independence, academic and professional integrity, consistent dedication to the development of new ideas or processes in advanced contexts of professional and scientific activity.			5%
PH4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість / Responsibly treat the works performed, be responsible for their quality			5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання / Correspondence between learning results and program study results

Програмні результати навчання	Результати навчання дисципліни									
	PH 1.1	PH 1.2	PH 1.3	PH 1.4	PH 2.1	PH 2.2	PH 2.3	PH 3.1	PH 4.1	PH 4.2
<i>(з опису освітньої програми)</i>										
ПРН-2. Знати принципи фінансування науково-дослідної роботи та структуру кошторисів на її виконання, вміння підготувати запит на отримання фінансування, звітну документацію / To know the principles of research funding and structure of estimates for its implementation, ability to prepare a request for funding, reporting documentation									+	+

7.2. Організація оцінювання:

Обов'язковим є виконання завдань, винесених на самостійну роботу, та контрольних робіт за графіком робочої програми.

У частину 1 входять теми 1 - 4, у частину 2 – теми 5 – 8. Обов'язковим для екзамену є виконання усіх контрольних робіт та захист проекту до вказаної викладачем дати, перед початком екзаменаційної сесії, згідно навчального плану. Переписування чи перескладання тем не практикується. Дозволяється здача окремих завдань модульних тем у проміжках між написанням модульних контрольних робіт (наприклад, перша тема здається до здачі наступної модульної контрольної роботи у будь-який зручний для викладача та здобувача час).

Терміни проведення форм оцінювання:

1. *Контрольна робота: до 5 тижня навчального періоду.*
2. *Контрольна робота: до 13 тижня навчального періоду.*
3. *Захист проекту: до 10 тижня навчального періоду.*

У випадку відсутності з поважних причин відпрацювання та перездачі контрольні роботи здійснюються у відповідності до „Положення про організацію освітнього процесу”.

7.3. Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

При визначенні оцінки визначальною є робота в семестрі. Після завершення розгляду тем проводяться письмові контрольні роботи та теоретичне опитування.

**8. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ. ТЕМАТИЧНИЙ ПЛАН ЛЕКЦІЙ
/ STRUCTURE OF THE DISCIPLINE. THEMATIC PLAN OF LECTURES**

№	Назва лекції	Кількість годин		
		Лекції	Практичні	Самостійна робота
Частина 1. «Основні методи формалізації алгоритмів та програмних систем»/ Part 1. “Main methods of formalization of algorithms and software systems				
1	Тема 1. Вступ, основні означення та поняття. Абстрактна обчислюваність. <i>Самостійна робота:</i> На прикладі задач по темі дисертаційного дослідження провести аналіз використовуваних алгоритмів. / Theme 1. Introduction, basic definitions and notions. Abstract computability. <i>Individual work:</i> Analyze the algorithms used in a dissertation research.	2		6
2	Тема 2. Багатоосновні алгебри. Моделювання предметної області за допомогою формалізма багатоосновних алгебр. <i>Самостійна робота:</i> На прикладі задач по темі дисертаційного дослідження провести побудову моделей предметної області. / Theme 2. Multisorted algebras. Modeling a subject domain using the formalism of multisorted algebras. <i>Individual work:</i> Construct models of subject domains used in a dissertation research.	2		6
3	Тема 3. Програмні алгебри та програмні логіки. <i>Самостійна робота:</i> Сформулювати по темі дисертаційного дослідження опис систем в програмній алгебрі. / Theme 3. Program algebras and program logics. <i>Individual work:</i> Formulate in software algebra the description of systems used in a dissertation research.	2		12
4	Тема 4. Методи аналізу програмних систем. / Theme 4. Methods of analysis of program systems.	1		6
	<i>Контрольна робота 1/ Task 1.</i>	1		
Частина 2. «Основні методи та технології розробки програмних систем»				
5	Тема 5. Методи специфікації програмних систем (B, TLA, Dafny). <i>Самостійна робота:</i> Написати специфікацію фрагмента програмної системи, пов'язаної із темою дисертаційного дослідження, за допомогою однієї з мов	2		12

	специфікацій. / Theme 5. Methods of program systems specification (B, TLA, Dafny). <i>Individual work:</i> Using a specification language write a specification of a fragment of program system related to the topic of a dissertation research.			
6	Тема 6. Формалізація вимог до програмних систем за допомогою програмних логік. <i>Самостійна робота:</i> Сформулювати по темі дисертаційного дослідження специфікацію фрагмента системи в програмній алгебрі. / Theme 6. Formalization of requirements to program systems using program logic. <i>Individual work:</i> Formulate specification in program algebra of a fragment of software system used in a dissertation research.	2		12
7	Тема 7. Методи верифікації програмних систем з використанням програмних логік. <i>Самостійна робота:</i> Верифікувати фрагмент програмної системи, пов'язаної із темою дисертаційного дослідження, з використанням програмної логіки. / Theme 7. Methods of verification of program systems using program logic. <i>Individual work:</i> Verify with program logic a fragment of software system used in a dissertation research.	2		12
8	Тема 8. Перспективні напрями створення методів та технологій розробки програмних систем. / Theme 8. Prospective directions of creation of methods and technologies of software systems development.	1		6
	<i>Контрольна робота 2/</i> <i>Task 2</i>	1		
	ВСЬОГО/ TOTAL	16		72

Загальний обсяг 90 годин, в тому числі / **Total duration 90 hours**, namely:

Лекції/Lectures – **18 годин/ hours**,

Консультації/ Consultations - **2 години/ hours**.

Самостійна робота / Individual work – **72 години/ hours**.

9. Рекомендовані джерела/ Recommended sources

Основні / Main:

1. С.Л. Кривий. Вступ до методів створення програмних систем. Київ, НаУКМА, 2018.–449.
2. Дорошенко А.Ю., Жереб К.А., Иванов Е.В., Никитченко Н.С., Яценко Е.А. Формальные методы построения параллельных программ, Кропивницький, 2016.– 440 с.

Додаткові / Additional:

3. Abrial J.A. Assigning Programs to Meanings. Cambridge: Cambridge University Press, 2005.– 816 p.
4. Bjorner D. Software Engineering 1: Abstraction and Modelling. Berlin: Springer, 2006.– 714 p.
5. Bjorner D. Software Engineering 2: Specification of Systems and Languages. Berlin: Springer, 2006.– 780 p.
6. Bjorner D. Software Engineering 3: Domains, Requirements, and Software Design. Berlin: Springer, 2010.– 768 p.
7. Lamport L. Specifying Systems: The TLA+ Language and Tools for Hardware and Software Engineers. Addison-Wesley Professional, 2002.– 384 p.
8. Leino, R. Dafny: An Automatic Program Verifier for Functional Correctness. Proceedings of the Conference on Logic for Programming, Artificial Intelligence, and Reasoning. 2010, pp. 348–370.