

Комп'ютерна математика (частина 3)

СИЛАБУС

Шифр і назва спеціальності	121 "Інженерія програмного забезпечення"	Інститут / факультет	Факультет комп'ютерних наук і програмної інженерії
Назва програми	«Інженерія програмного забезпечення»	Кафедра	Кафедра програмної інженерії та інформаційних технологій управління
Тип програми	Освітньо-професійна	Мова навчання	Українська, англійська

Викладач

ПІБ, електронна пошта

Хацько Наталя Євгенівна, nataliia.khatsko@khpі.edu.ua



К.т.н., доцент кафедри програмної інженерії та інформаційних технологій управління, доцент.

Підготувала і опублікувала понад 45 наукових та навчально-методичних публікацій.

SCOPUS Author ID <https://www.scopus.com/authid/detail.uri?authorId=57200820629>; Researcher ID

<https://app.webofknowledge.com/author/#/record/17252627>; Google Scholar

<https://scholar.google.com.ua/citations?user=US7Ovx4AAAAAJ&hl=uk>; ORCID orcid.org/0000-0002-2543-0280

Основні курси: «Комп'ютерна математика (частини 1, 2, 3)», «Практичний семінар з математичних методів в інженерії програмного забезпечення», «Формальні методи верифікації програмних систем», «Формальні методи дослідження програмних систем».

Загальна інформація про курс

Анотація	Дисципліна «Комп'ютерна математика (частина 3)» є навчальною дисципліною з циклу професійної обов'язкової підготовки за спеціальністю 121 "Інженерія програмного забезпечення". Вона викладається у п'ятому семестрі в обсязі 180 год. (6 кредитів ECTS), зокрема: лекції – 32 год., практичні заняття – 32 год., самостійна робота – 116 год. У курсі передбачено два змістових модулі та дві модульні контрольні роботи. Завершується дисципліна екзаменом.
Цілі курсу	Формування у студентів теоретичних та практичних знань щодо теорії скінченних автоматів та супроводжуваних її понять (граматика, мова, регулярний вираз). Формування у студентів сучасної системи поглядів у галузі комп'ютерної дискретної математики, набуття практичних навичок щодо використання формальних методів і моделей дискретної математики при обробці дискретної інформації та опису дискретних процесів, пов'язаних з розробкою програмного забезпечення.
Формат	Лекції, практичні роботи, самостійна робота. Підсумковий контроль — іспит.
Семестр	5

Обсяг (кредити) / Тип курсу
(обов'язковий / вибірковий)

6 / обов'язковий

Лекції (години)

32

Практичні заняття (години)

32

Самостійна робота (години)

116

Програмні компетентності

K01. Здатність до абстрактного мислення, аналізу та синтезу.

K05. Здатність вчитися і оволодівати сучасними знаннями.

K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

K26. Здатність до алгоритмічного та логічного мислення.

Результати навчання	Методи викладання та навчання	Форми оцінювання (поточне оцінювання CAS, підсумкове оцінювання FAS)
ПРО1. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	Лекції, практичні заняття, виконання домашніх завдань	Поточне оцінювання CAS: Оцінювання роботи студентів у процесі практичних занять Тестування Підсумкове оцінювання FAS: Іспит
ПРО5. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення		

СИСТЕМА ОЦІНЮВАННЯ

Розподіл балів для оцінювання успішності аспіранта	Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	Нарахування балів	100% підсумкове оцінювання у вигляді іспиту (40%) та поточного оцінювання (60%). 40% іспит ... 60% поточне оцінювання: - 40% оцінювання завдань на практичних роботах; - 20% проміжний контроль
	90-100	A	відмінно		
	82-89	B	добре		
	74-81	C			
	64-73	D	задовільно		
	60-63	E			
	35-59	FX	незадовільно з можливістю повторного складання		
	0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни		

Політика курсу

Структура та зміст курсу

Лекція 1-2	Введення в теорію формальних доказів	Практичне заняття 1-2	Дедуктивні методи доказів. Побудова доказів.	Самостійна робота	Вивчення матеріалу за допомогою літератури, виконання домашніх завдань.
Лекція 3-4	Алфавіти, граматики та мови	Практичне заняття 3-4	Розв'язання задач виводу ланцюжків по правилам виводу, розбір ланцюжків		
Лекція 5	Регулярні вирази, мови.	Практичне заняття 5	Модульна контрольна робота.		
Лекція 6-7	Детерміновані скінченні автомати	Практичне заняття 6-7	Розв'язання задач. Опис детермінованих скінченних автоматів.		

Лекція 8-9	Недетерміновані скінченні автомати	Практичне заняття 8-9	Розв'язання задач. Опис недетермінованих скінченних автоматів, та їх перетворення.	
Лекція 10-11	Скінченні автомати з епсілон-переходами	Практичне заняття 10-11	Перетворення автоматів.	
Лекція 12-13	Скінченні автомати та регулярні вирази	Практичне заняття 12-13	Перетворення детермінованого скінченного автомату в регулярний вираз.	
Лекція 14-15	Перетворення регулярного виразу в скінченний автомат	Практичне заняття 14-15	Перетворення регулярного виразу в скінченний автомат.	
Лекція 16	Автомати з магазинною пам'яттю	Практичне заняття 16	Модульна контрольна робота	

Література

1. Ajit Singh (2019) Formal Language And Automata Theory.
2. Abejide Ade-Ibijola (2017) New Finite Automata Applications in Novice Program Comprehension. LAP LAMBERT Academic Publishing, 2017.
3. Neeru Gupta (2020) Beginner's Guide - Automata Theory.
4. Ezhilarasu Umadevi Palani (2019) Finite Automata Problems & Solutions. LAP Lambert Academic Publishing.
5. Stoyan Mihov, Klaus U. Schulz (2019) Finite-State Techniques: Automata, Transducers and Bimachines. - Cambridge University Press.
6. Hopcroft, John E., Motwani, Rajeev; Ullman, Jeffrey D.(2014) Introduction to automata theory, languages, and computation Boston : Pearson Education, Inc.
7. Хаггарти Р.(2017) Дискретная математика для программистов: учебное пособие. Москва : Техносфера,.

1. Michael Sipser (2006) Introduction to the Theory of Computation Thomson.
2. Daniel I. A. (1996) Cohen Introduction to Computer Theory, / Daniel I.A. Cohen, John Wiley.
3. John C Martin, Lewis H.P. C.H. Pearson Introduction to languages and the Theory of Computation.
4. Elements of Theory of Computation”, & Papadimition.
5. Theory of Computer Science and Automata languages and computation Mishra and Chandrashekaran, 2nd edition.
6. Foster, E. C. Software Engineering : A Methodical Approach. New York: Apress.
7. J. Richard Büchi (1989) Finite Automata, Their Algebras and Grammars. Towards a Theory of Formal Expressions.
8. Goswami D.. Krishna K. V. (2010) Formal Languages and Automata Theory.
9. Завалишин Е.П. (2007) Логика. Учебное пособие для вузов. Тула: Изд-во ТулГУ
10. Gerda Ivanickienė. The theoretical material and exercises.

ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ

1. D. Goswami and K. V. Krishna. Formal Languages and Automata Theory:
<http://www.iitg.ernet.in/dgoswami/Flat-Notes.pdf>
2. Introduction to Automata and Complexity Theory:
<http://infolab.stanford.edu/~ullman/ialc/spr10/spr10.html>
3. Formal Languages and Automata Theory:
<http://cs.fit.edu/~dmitra/FormaLang>.

Норми академічної етики

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність

Силабус за змістом повністю відповідає робочій програмі курсу.