

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи
Олена КАШПУР
«07» травня 2021 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Теорія обчислень та комп'ютерні технології

для студентів

галузь знань **12 «Інформаційні технології»**
спеціальність **122 «Комп'ютерні науки»**
освітній рівень **магістр**
освітня програма **«Інформатика»**
вид дисципліни **обов'язкова**

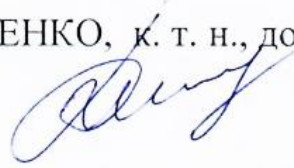
Форма навчання	денна
Навчальний рік	2021/2022
Семестр	1
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: к.т.н., доц. **Олексій ТКАЧЕНКО** (лекції, лабораторні заняття)
к.ф.-м.н., доц. **Тарас ПАНЧЕНКО** (лабораторні заняття)

Пролонговано: на 20__ / 20__ н.р. _____ (_____) «__» 20__ р.
на 20__ / 20__ н.р. _____ (_____) «__» 20__ р.

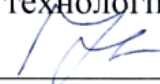
КИЇВ – 2021

Розробник: Олексій ТКАЧЕНКО, к. т. н., доцент, доцент кафедри теорії та технології програмування.



ЗАТВЕРДЖЕНО

Завідувач кафедри
теорії та технології програмування

 Микола НІКІТЧЕНКО

Протокол № 10 від « 27 » квітня 2021 року

Схвалено гарантом освітньо-наукової програми «Інформатика»

 Степан ШКІЛЬНЯК

« 6 » травня 2021 року

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

Протокол від « 6 » травня 2021 року № 10

Голова науково-методичної комісії  Людмила ОМЕЛЬЧУК

« 6 » травня 2021 року

1. Мета дисципліни – оволодіння знаннями в області теорії обчислень, засвоєння умінь і навичок, використання засвоєних знань при реалізації етапів життєвого циклу програмних систем, зокрема, імплементації та перевірки коректності алгоритмів із застосуванням сучасного програмного інструментарію, технологій і підходів програмної інженерії.

2. Попередні вимоги до опанування або вибору навчальної дисципліни "Теорія обчислень та комп'ютерні технології". Студент повинен опанувати основи програмування, математичної логіки та теорії алгоритмів, знати формальні методи розробки програм.

3. Анотація навчальної дисципліни:

Навчальна дисципліна "Теорія обчислень та комп'ютерні технології" є складовою освітньо-наукової програми підготовки фахівців за *освітнім рівнем* "магістр" *галузі знань* 12 "Інформаційні технології" за *спеціальністю* 122 "Комп'ютерні науки", *програми* "Інформатика".

Викладається у 1-му семестрі в **обсязі 150 годин (5 кредитів ECTS)**, зокрема: *лекції – 26 год., лабораторні – 22 год., консультації – 2 год., самостійна робота – 100 год.* В курсі передбачено *2 частини та 2 контрольні роботи.*

Завершується дисципліна **іспитом у 1-му семестрі.**

У результаті вивчення навчальної дисципліни студент повинен:

знати: основні концепції класичних і некласичних моделей обчислення, інтенціональні та екстенціональні аспекти, а також ключові проблеми теорії обчислюваності, особливості реалізації композиційного, функціонального, логічного, компоненто-орієнтованого підходів у програмуванні, проблематику і технології інтеграції програмного забезпечення;

вміти: аналізувати і створювати програми, що реалізують алгоритми класичних моделей обчислень, а також композиційний, функціональний, логічний підходи у програмуванні.

Місце дисципліни. Навчальна дисципліна "Теорія обчислень та комп'ютерні технології" є складовою циклу професійної підготовки фахівців освітнього ступеня "Магістр" освітньої програми "Інформатика".

4. Завдання (навчальні цілі): набуття знань, умінь та навичок (компетенцій) на рівні новітніх досягнень у програмуванні, відповідно до освітньої кваліфікації "Магістр з комп'ютерних наук" за спеціалізацією "Теорія та технологія програмування".

Зокрема, розвивати:

– Здатність спілкуватися іноземною мовою (ЗК-5);
– Здатність розробляти й управляти проектами (ЗК-11);
– Здатність застосовувати квантові операції, виміри, алгоритми для розв'язання задач, пов'язаних з особливостями реалізації квантових обчислень в різних фізичних системах, зокрема в квантовій криптографії (СК-7);

– Здатність аналізувати сучасні світові тенденції розвитку комп'ютерних наук та уявляти перспективи розвитку інформаційних технологій, моделювати процеси розвитку і трансформації інформаційно- комунікаційних технологій в практичній професійній роботі (СК-11);

– Здатність проводити дослідження функціональної та економічної ефективності та надійності інформаційних систем (СК-13);

– Здатність проектування динамічних веб-додатків як інформаційної системи із застосуванням об'єктно-орієнтованих технологій програмування, зокрема сучасних програмних засобів підтримки взаємодії клієнта та сервера із застосуванням розподілених систем керування базами даних, супроводження та оптимізація веб-сторінок (СК-14).

5. Результати навчання за дисципліною

Результат навчання (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			
РН1.1	Знати основні положення класичних моделей обчислень, інтенціональні та екстенціональні аспекти, основні проблеми теорії обчислювальності.	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, Іспит	20%
РН1.2	Знати основні концепції квантових обчислень, а також композиційного, функціонального, логічного, компонентно-орієнтованого підходу у програмуванні.	Лекція, лабораторне заняття, доповідь	Тест, 60% правильних відповідей, Іспит	15%
РН1.3	Знати особливості архітектури складних програмних систем, технології взаємодії віддалених компонентів та інтеграції програмного забезпечення.	Лекція, лабораторне заняття, доповідь	Тест, 60% правильних відповідей, Іспит	15%
РН2.1	Вміти аналізувати і створювати програми, що реалізують сучасні парадигми програмування.	Лабораторне заняття, самостійна робота	Доповідь, захист лабораторної роботи, Іспит	32%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Лабораторне заняття, самостійна робота	Доповідь, захист лабораторної роботи	10%
РН4.1	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лабораторна робота, самостійна робота	Захист лабораторної роботи	8%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	РН 1.1	РН 1.2	РН 1.3	РН 2.1	РН 3.1	РН 4.1
Програмні результати навчання						
<i>(з опису освітньої програми)</i>						
ПРН-5. Вирішувати складні проблеми, що вимагають систем з великою обчислювальною потужністю для забезпечення масштабованості паралельних алгоритмів і програм.	+		+	+	+	
ПРН-6. Використовувати розподілені високопродуктивні обчислювальні технології для забезпечення ефективного вибору та використання консолідованих ресурсів і послуг				+		+
ПРН-8. Аналізувати особливості використання сучасних квантових технологій для забезпечення вирішення проблем, зокрема конфіденційного зв'язку, квантової криптографії, здійснювати дослідження теоретичних та експериментальних аспектів квантової інформатики		+			+	+

7. Схема формування оцінки

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота (тест) 1: РН 1.1, РН1.2 – 15 балів/9 балів.
2. Контрольна робота (тест) 2: РН1.2, РН1.3 – 15 балів/9 балів.
3. Підготовка доповіді (за темами лекційних занять та самостійної роботи): РН 1.1, РН1.2, РН 1.3, РН 3.1 – 15 балів/9 балів.
4. Лабораторна робота (проект): РН 2.1, РН 3.1, РН 4.1 – 15 балів/9 балів.

- підсумкове оцінювання (у формі іспиту):

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;
- результати навчання які будуть оцінюватись: РН 1.1 – РН 1.3, РН 2.1
- форма проведення і види завдань: письмова форма

Види завдань:

Структура екзаменаційної роботи та критерії оцінювання:

1. Теоретичне запитання (РН 1.1).
2. Теоретичне запитання (РН 1.2 – РН 1.3).
3. Письмове практичне завдання (РН 2.1).
4. Письмове практичне завдання (РН 2.1).

Критерії оцінювання екзаменаційної роботи

Завдання	Вид завдання	Максимальний бал (відсоток)	Всього балів (відсотків)
Завдання 1, 2	Теоретичне запитання	по 10 балів (25 %)	20 балів (50 %)
Завдання 3	Письмове практичне завдання	10 балів (25 %)	10 балів (25 %)
Завдання 4	Письмове практичне завдання	10 балів (25 %)	10 балів (25 %)
Всього			40 балів (100%)

Запитання для підготовки до іспиту

1. Екстенсіональні та інтенсіональні аспекти теорії обчислюваності.
2. Класичні та неklasичні моделі обчислень.
3. Іntenсіональні дані, їх формалізація.
4. Загальна схема інтенсіональної обчислюваності.
5. Класи інтенсіонально-обчислювальних функцій: основні результати та схема їх доведення.
6. Теорія складності обчислень. Класи P і NP.
7. Приклади проблем класу P.
8. Приклади проблем класу NP.
9. Додаткові класи проблем. Обчислюваність у поліноміальному просторі.
10. Рандомізація в обчисленнях.
11. Модулярна арифметика. Складність обчислень.
12. Лямбда-числення. Математичні основи та використання в сучасних мовах програмування.

13. Теоретичні засади логічного програмування. Логіка висловлень.
14. Методи доведення істинності та виконаності формул.
15. Логіка предикатів першого порядку як основа логічного програмування. Резолюції. Приклад реалізації.
16. Композиційний підхід в сучасному програмуванні.
17. Рекурсивні функції та обчислення. Математичні основи, особливості реалізації.
18. Реалізація функціональної парадигми в сучасних засобах розробки ПЗ.
19. Реалізація логічного програмування в сучасних засобах розробки ПЗ.
20. Багатозначна логіка та її реалізація у програмуванні.
21. Рефлексії у програмуванні.
22. Архітектура великих обчислювальних систем.
23. Компонентно-орієнтоване програмування. Властивості компонентів. Модель посилань.
24. Модульність. Специфікація інтерфейсу компоненту.
25. Взаємодія компонентів. Обмін повідомленнями.
26. Взаємодія компонентів. Віддалений виклик процедур.
27. Взаємодія віддалених об'єктів. Маршалінг, серіалізація.
28. Приклади проектування компонентів.
29. Проблеми, стратегії і технології інтеграції програмного забезпечення.
30. Квантові обчислення. Специфіка. Реалізація.
31. Теорія обчислень та формалізація життєвого циклу ПЗ.

Для отримання загальної позитивної оцінки з дисципліни оцінка за іспит не може бути меншою 24 балів

Студент не допускається до іспиту, якщо під час семестру набрав менше ніж 36 балів.

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. *Контрольна робота (тест): до 7 тижня семестру.*
2. *Контрольна робота (тест): до 14 тижня семестру.*
3. *Підготовка доповіді (за темами лекційних занять та самостійної роботи): до 10 тижня семестру.*
4. *Лабораторна робота (проект): до 14 тижня семестру.*

Студент має право на одне перескладання контрольної роботи із можливістю отримання максимально 10 балів за кожну. Термін перескладання визначається викладачем.

У разі неякісного виконання лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторну роботу та доповідь після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

7.3 Шкала відповідності оцінок

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

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лаб. роб.	Сам. р-та
Частина 1. Сучасні проблеми теорії обчислень				
1	Теорія обчислюваності на сучасному етапі. <i>Самостійна робота:</i> Основні концепції обчислень та обчислюваності. Робота з літературними джерелами	2		8
2	Класичні та неklasичні моделі обчислень. Екстенціональні та інтенціональні аспекти теорії обчислюваності. <i>Самостійна робота:</i> Методи парсингу обчислювальних виразів. Програмний проєкт парсингу обчислювальних виразів.	2	2	8
3	Інтенціональні дані, їх формалізація. Загальна схема інтенціональної обчислюваності. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	2	8
4	Класи інтенціонально-обчислювальних функцій: основні результати та схема їх доведення. <i>Самостійна робота:</i> Високопродуктивні розподілені обчислення. Лінійні та інтенсивні обчислення. Виконання лабораторної роботи.	2	2	8
5	Класи проблем теорії обчислень. <i>Самостійна робота:</i> Квантові обчислення та їх застосування. Виконання лабораторної роботи.	1	2	6
	Контрольна робота 1	1		
	Всього за частиною 1	10	8	38
Частина 2. Технології обчислень				
6	Композиційний підхід в сучасних засобах розробки ПЗ. <i>Самостійна робота:</i> Еволюція підходів до управління життєвим циклом ПЗ. Коректність програми, методи її доведення. Підходи забезпечення якості ПЗ. Виконання лабораторної роботи.	2	2	8
7	Реалізація лямбда-числення та функціонального підходу в сучасних мовах програмування. <i>Самостійна робота:</i> Застосування функціонального підходу. Імплементация лямбда-числення в Java, C#, Python. Виконання лабораторної роботи.	2	2	8
8	Реалізація рекурсивних обчислень в сучасних мовах програмування. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	2	8
9	Реалізація логічного підходу в сучасних засобах розробки ПЗ. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	2	8
10	Метадані. Рефлексії у програмуванні. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	2	8

11	Компоненто-орієнтоване програмування. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	2	8
12	Технології реалізації віддаленої взаємодії програмних компонентів. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	2	1	8
13	Проблеми і технології інтеграції програмного забезпечення. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.	1	1	6
	Контрольна робота 1	1		
	Всього за частиною 2	16	14	62
	ВСЬОГО	26	22	100

Загальний обсяг 150 год., в тому числі:

Лекцій – **26 год.**

Лабораторні заняття – **22 год.**

Консультації – **2 год.**

Самостійна робота – **100 год.**

9. Рекомендовані джерела

Основні:

1. Нікітченко М.С., Шкільняк С.С. Прикладна логіка. – К.: ВПЦ Київський ун-т, 2013.
2. Нікітченко М.С., Шкільняк С.С. Математична логіка та теорія алгоритмів. – К.: ВПЦ Київський ун-т, 2008.
3. N. Nikitchenko. Intensional aspects of the notion of program. *Problems of Programming*, No. 3-4 (2001). 5–13.
4. Nikitchenko, M., Chentsov, A. Basics of Intensionalized Data: Presets, Sets, and Nominats. *Computer Science Journal of Moldova*, vol.20, no.3(60), 2012. 334–365.
5. Кривий С.Л. Дискретна математика. Підручник для студентів вищих навчальних закладів / С.Л. Кривий. – Вид. 2-ге. – Чернівці-Київ: Видавничий дім "Букрек", 2017.

Додаткові:

6. Хопкрофт, Джон, Э., Мотвани, Раджив, Ульман, Джеффри, Д. Введение в теорию автоматов, языков и вычислений, 2-е изд.: Пер. с англ. – М. : Издательский дом "Вильямс", 2008. — 528 с.
7. Стюарт Т. Теория вычислений для программистов / Пер. с англ. А. А. Слипкин. - М.: ДМК Пресс, 2014. - 384 с.
8. Шкільняк С.С. Теорія алгоритмів. Приклади й задачі. – К.: ВПЦ Київський ун-т, 2012.
9. Maheshwari A., Smid M. Introduction to Theory of Computation. – Ottawa, 2019, 244 p.
10. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. (пер. с англ.) – М., 1979. – 536 с.