

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

КАФЕДРА МАТЕМАТИЧНОЇ ІНФОРМАТИКА



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
СКЛАДНІСТЬ АЛГОРИТМІВ
для студентів

галузь знань	12 «Інформаційні технології» <i>(шифр і назва)</i>	
спеціальність	122 «Комп'ютерні науки» <i>(шифр і назва спеціальності)</i>	
освітній рівень	бакалавр <i>(молодший бакалавр, бакалавр, магістр)</i>	
освітня програма	«Інформатика» <i>(назва освітньої програми)</i>	
вид дисципліни	вибіркова	
вибірковий блок	«Інтелектуальні інформаційні технології»	
	Форма навчання	денна
	Навчальний рік	2021/2022
	Семестр	7
	Кількість кредитів ECTS	3
	Мова викладання, навчання та оцінювання	українська
	Форма заключного контролю	залік

Викладачі: **магістр, асистент Федорус О.М.** (лекції),

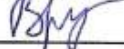
Пролонговано: на 20__/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

Розробник: **Федорус Олексій Мстиславович**, асистент, кафедра математичної інформатики
Терещенко Василь Миколайович, д.ф.-м.н., професор, завідувач кафедри математичної інформатики

ЗАТВЕРДЖЕНО

Завідувач кафедри математичної інформатики

 Василь ТЕРЕЩЕНКО

Протокол № 6 від «11» 02 2021 р.

Схвалено Гарантом освітньо-професійної програми «Інформатика»

«11» лютого 2021р.  Людмила ОМЕЛЬЧУК

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

Протокол від «11» лютого 2021 року № 7

Голова науково-методичної комісії  Людмила ОМЕЛЬЧУК

«11» лютого 2021 року

1. Мета дисципліни "Складність алгоритмів" є набуття теоретичних та практичних знань в одній з найактуальніших на сьогодні галузей інформаційних технологій: аналіз алгоритмів та визначення їх складності. У ході навчання студенти ознайомляться з основними моделями для аналізу алгоритмів та визначенню їх складності.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. Успішне опанування курсів «Програмування».

2. Володіти сучасними програмними мовами, засобами розробки та проектування програм для розв'язання наукових та прикладних задач та навичками навчальної діяльності.

3. Анотація навчальної дисципліни:

Навчальна дисципліна «Командна розробка програмного продукту» є складовою освітньо-професійної програми підготовки фахівців за першим (бакалавр) рівнем вищої освіти галузі знань 12 „Інформаційні технології” зі спеціальності 122 „Комп’ютерні науки”, освітньо-професійної програми „Інформатика”.

Дана дисципліна є дисципліною за вибором для спеціальності **“Інформатика”**.

Викладається у 1 семестрі 4 курсу в обсязі –42 год.

(3 кредити ECTS) зокрема: лекції – 40 год., консультації – 2 год. У курсі передбачено 2 змістовних модулів. Завершується дисципліна – заліком в 1 семестрі.

В результаті вивчення навчальної дисципліни студент повинен:

Знати основні моделі для аналізу алгоритмів та визначенню їх складності.

Вміти провести аналіз алгоритму та оцінити його складність.

Для допуску до дисципліни «Складність алгоритмів» освітньо-професійної програми «Інформатика» студент повинен опанувати компетентності та результати навчання, які надає дисципліни “Дискретна математика”, “Програмування”, ”Теорія алгоритмів та математична логіка”. програми «Інформатика».

4. Завдання (навчальні цілі):

набуття теоретичних та практичних знань в одній з найактуальніших на сьогодні галузей інформаційних технологій: аналіз алгоритмів та визначення їх складності. У ході навчання студенти ознайомляться з основними моделями для аналізу алгоритмів та визначенню їх складності.

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	Методи оцінювання та пороговий критерій	Відсоток у підсумковій оцінці з дисциплін
Код	Результат навчання			

		навчання	оцінювання (за необхідності)	и
PH1.1	<i>Знати моделі для аналізу та оцінки складності алгоритмів</i>	<i>Лекція, самостійні завдання</i>	<i>Тест, 60% правильних відповідей</i>	16%
PH1.2	<i>Знати складності основних алгоритмів</i>	<i>Лекція, самостійні завдання</i>	<i>Тест, 60% правильних відповідей</i>	15%
PH1.3	<i>Знати основні підходи до проектування великих систем</i>	<i>Лекція, лабораторне заняття</i>	<i>Тест, 60% правильних відповідей</i>	20%
PH2.1	<i>Вміти провести аналіз алгоритму та оцінити його складність</i>	<i>Самостійна робота, лекція</i>	<i>Перевірка домашніх завдань</i>	23%
PH3.1	<i>Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програмних систем, складати письмові звіти</i>	<i>Самостійна робота</i>	<i>Поточне оцінювання, перевірка домашніх завдань</i>	10%
PH4.1	<i>Організовувати свою самостійну роботу для досягнення результату</i>	<i>Самостійна робота</i>	<i>Поточне оцінювання</i>	8%
PH4.2	<i>Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість</i>	<i>Самостійна робота</i>	<i>Перевірка домашніх завдань</i>	8%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	PH	PH	PH	PH	PH	PH	PH
	1.1	1.2	1.3	2.1	3.1	4.1	4.2
Програмні результати навчання							
<i>(з опису освітньої програми)</i>							
ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність 10 алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.	+	+	+	+	+	+	+

7. Схема формування оцінки.

7.1. Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота 1: PH1.1, PH 1.2 - 40 балів/24 бали

2. Контрольна робота 2: PH1.3 - 40 балів/24 бали

3. Домашня письмова робота: 10 балів/6 балів

4. Усна відповідь, доповнення: по 5 балів/3 бали в 2х частинах

- підсумкове оцінювання у формі заліку. Виставляється за результатами роботи студентами впродовж усього семестру та не передбачає додаткових заходів оцінювання для успішних студентів.

Запитання для підготовки до контрольних робіт

1. Параметри оцінки складності алгоритмів.
2. Особливості моделі ДАМ.
3. Модель ДАМП та порівняння з моделлю ДАМ.
4. Багатострічкова машина Т'юринга.
5. Методика оцінки складності програм на мовах високого рівня.
6. Оцінка складності рекурсивних програм.
7. Алгоритми на графах та оцінки їх складності.
8. Оцінка складності для множення матриць за алгоритмом Штрассена.
9. Обернення матриць.
10. НВП-розклад матриць.
11. Недетермінована машина Т'юринга.
12. NP-повні задачі.
13. Приклади NP-повних задач.

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. Контрольна робота (тест): до 7 тижня семестру.
2. Контрольна робота (тест): до 15 тижня семестру.

Студент має право на одне перескладання кожної контрольної роботи із можливістю отримання максимально 80% початково визначених за цю контрольну роботу балів. Термін перескладання визначається викладачем.

7.3 Шкала відповідності оцінок

Зараховано / Passed	60-100
Не зараховано / Fail	0-59

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

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабор. заняття	Самост. робота
Частина 1				
Моделі для аналізу та оцінки складності алгоритмів				
1	Тема 1. Поняття про складність алгоритмів. Моделі ДАМ і ДАМП та зв'язок між ними. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		6
2	Тема 2. Багатострічкова машина Т'юринга. Зв'язки з моделями ДАМ і ДАМП. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		6
3	Тема 3. Мови високого рівня та оцінки складності програм. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	6		6
4	Тема 4. Множини, їх представлення. Оцінки складності для операцій над множинами. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		6
5	Тема 5. Рекурсія. Методика оцінки складності рекурсивних алгоритмів. Динамічне програмування. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		6
6	Тема 6. Алгоритми сортування. Оцінки їх складності. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		6
7	Тема 7. Алгоритми на графах. Остовне дерево найменшої вартості. Пошуки в глибину та в ширину. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		4
контрольна робота		2		
Частина 2				
Алгоритми для дій над матрицями. NP-повні задачі.				
8	Тема 8. Множення та обернення матриць. Алгоритм Штрассена. НВП-розклад матриць. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		4
9	Тема 9. Недетермінована машина Т'юринга. NP-повні задачі. Приклади. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		4
контрольна робота		2		
ВСЬОГО		42		48

Загальний обсяг 42 години, в тому числі:

Лекцій – **40 годин**.

Консультації – **2 год**.

Самостійна робота - **48 год**.

9. Рекомендовані джерела:

Основні

1. Анисимов А.В. Рекурсивные преобразователи информации. 1987.
2. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. Мир. 1979.
3. Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Мир. 1980.

Додаткові

4. Грэхем Р., Кнут Д., Паташник О. Конкретная математика. Основание информатики. Мир. 1998. – 703с.