КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

КАФЕДРА МАТЕМАТИЧНОЇ ІНФОРМАТИКА

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи

_____ Олена КАШПУР

«12» лютого 2021 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ СКЛАДНІСТЬ АЛГОРИТМІВ

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **122 «Комп'ютерні науки»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Інформатика»**
(назва освітньої програми)
вид дисципліни **вибіркова**
вибірковий блок **"Інтелектуальні інформаційні технології"**

Форма навчання	**денна**
Навчальний рік	**2023/2024**
Семестр	**7**
Кількість кредитів ECTS	**4**
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)

КИЇВ – 2021

Розробник: **Федорус Олексій Мстиславович**, асистент, кафедра математичної інформатики Фед

Терещенко Василь Миколайович, д.ф.-м.н., завідувач кафедри математичної інформатики

ЗАТВЕРДЖЕНО
Зав. кафедри математичної інформатики

_____ Василь ТЕРЕЩЕНКО
(підпис) (прізвище та ініціали)

Протокол № 8 від «11» 02 2021 р.

Схвалено гарантом освітньо-професійної програми «Інформатика»

_____ Людмила ОМЕЛЬЧУК «11» лютого 2021 рік
(підпис) (прізвище та ініціали)

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

Протокол від «11» лютого 2021 року № 7

Голова науково-методичної комісії _____ Людмила ОМЕЛЬЧУК

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 ознайомлюються 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 «Складність алгоритмів» є складовою вибіркового блоку "Інтелектуальні інформаційні технології" освітньо-професійної програми "Інформатика" підготовки фахівців за першим (*бакалавр*) рівнем вищої освіти *галузі знань* 12 „Інформаційні технології" зі *спеціальності 122* „Комп'ютерні науки".

(3 кредити ECTS) *зокрема: лекції – 42 год., консультації – 2 год, самостійна робота 76 год. У курсі передбачено 2 частини.* 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.

4. Завдання (навчальні цілі):

Набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у програмуванні та обробці великих об'ємом даних, відповідно освітньої кваліфікації «Бакалавр з комп'ютерних наук». Зокрема, розвивати здатність до алгоритмічного мислення.
Набуття теоретичних та практичних знань в одній з найактуальніших на сьогодні галузей інформаційних технологій: аналіз алгоритмів та визначення їх складності. У ході навчання студенти ознайомлюються 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 навчання**	**Методи оцінювання та пороговий критерій оцінювання (за необхідності)**	**Відсоток у підсумковій оцінці з дисциплини**
Код	**Результат навчання**			
РН1.1	*Знати моделі для аналізу та оцінки складності алгоритмів*	*Лекція, самостійна робота*	*Тест, 60% правильних відповідей*	16%

РН1.2	*Знати складності основних алгоритмів*	*Лекція, самостійна робота*	*Тест, 60% правильних відповідей*	15%
РН1.3	*Знати основні підходи до проектування великих систем*	*Лекція, самостійна роботая*	*Тест, 60% правильних відповідей*	20%
РН2.1	*Вміти провести аналіз алгоритму та оцінити його складність*	*Лекція, самостійна робота*	*Перевірка домашніх завдань*	23%
РН3.1	*Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програмних систем, складати письмові звіти*	*Лекція, самостійна робота*	*Поточне оцінювання, перевірка домашніх завдань*	10%
РН4.1	*Організовувати свою самостійну роботу для досягнення результату*	*Самостійна робота*	*Поточне оцінювання*	8%
РН4.2	*Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість*	*Самостійна робота*	*Перевірка домашніх завдань*	8%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни **Програмні результати навчання**	**РН 1.1**	**РН 1.2**	**РН 1.3**	**РН 2.1**	**РН 3.1**	**РН 4.1**	**РН 4.2**
(з опису освітньої програми)							
ПРН19.1. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	+	+	+	+	+	+	+

7. Схема формування оцінки.

7.1. Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота 1: РН1.1, РН 1.2 - 40 балів/24 бали

2. Контрольна робота 2: РН1.3 - 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 Штрасена.
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 I Моделі для аналізу та оцінки складності алгоритмів				
1-3	**Тема 1.** Поняття про складність алгоритмів. Моделі ДАМ і ДАМП та зв'язок між ними. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	6		8
4-5	**Тема 2.** Багатострічкова машина Т'юринга. Зв'язки з моделями ДАМ і ДАМП. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		8
6-7	**Тема 3.** Мови високого рівня та оцінки складності програм. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		8
8-9	**Тема 4.** Множини, їх представлення. Оцінки складності для операцій над множинами. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		8
10-11	**Тема 5.** Рекурсія. Методика оцінки складності рекурсивних алгоритмів. Динамічне програмування. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		8
12-13	**Тема 6.** Алгоритми сортування. Оцінки їх складності. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	4		9
14-15	**Тема 7.** Алгоритми на графах. Остовне дерево найменшої вартості. Пошуки в глибину та в ширину. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	3		9
Контрольна робота		1		
Частина II Алгоритми для дій над матрицями. NP-повні задачі.				
16-18	**Тема 8.** Множення та обернення матриць. Алгоритм Штрасена. НВП-розклад матриць. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	6		9
19-21	**Тема 9.** Недетермінована машина Т'юринга. NP-повні задачі. Приклади. *Самостійна робота:* опрацювання лекційного матеріалу. Реалізація алгоритмів.	5		9
Контрольна робота		1		

ВСЬОГО	42		76

Загальний обсяг 120 *години*, в тому числі:
Лекцій – **42** *годин.*
Консультації – ***2*** *год.*
Самостійна робота **-** ***76*** *год.*

9. РЕКОМЕНДОВАНІ ДЖЕРЕЛА:

Основні:

1. Cormen Thomas H., Leiserson Charles E., Rivest Ronald L., Stein Clifford. Introduction to Algorithms. – 3rd Edition. – MIT Press, 2009. – 1292 с.
2. Вергунова І.М. Побудова та аналіз алгоритмів. Лекції. – Вінниця: ТВОРИ, 2020. –164 с.
3. Donald E. Knuth. The Art of Computer Programming. Volume 3. Sorting and Searching. 2nd Edition. – Addison-Wesley Professional, 1998. – 800 с.
4. Jon Kleinberg, Éva Tardos. Algorithm Design, 1st Edition. – Courier Westford, 2006. – 864 с. https://edisciplinas.usp.br/pluginfile.php/7933913/course/section/6549987/Algorithm%20Design.pdf.

10. Додаткові ресурси:

1. Вергунова І.М. Побудова та аналіз алгоритмів. Лекції. – Вінниця: ТВОРИ, 2020. Ел. ресурс. Режим доступу: http://csc.knu.ua/media/filer_public/73/98/73985450-57a3-4373-ba2b-4d62e9e4a552/posibn_pobud_algor.pdf.