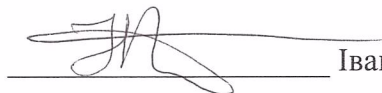


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіофізики та комп'ютерних технологій

Затверджено

На засіданні
кафедри радіофізики та комп'ютерних
технологій
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол №*72* від *6* *с* *г* *р* *м* *е* 2022 р.)

Завідувач кафедри:


Іван КАРБОВНИК

Силабус з навчальної дисципліни
“Алгоритми і структури даних”,
що викладається в межах ОПП
“ Високопродуктивний комп'ютинг ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2022 р.

Назва дисципліни	Алгоритми та структури даних
Адреса викладання дисципліни	м. Львів, вул. Драгоманова, 50 м. Львів, вул. Гарнавського, 107
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра радіофізики та комп'ютерних технологій
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології, 121 Інженерія програмного забезпечення
Викладачі дисципліни	Вельгош Сергій Романович, канд. фіз.-мат. наук, доцент
Контактна інформація викладачів	serhiy.velhosh@lnu.edu.ua https://electronics.lnu.edu.ua/employee/velhosh-s-r
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі он-лайн консультації через MS Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://teams.microsoft.com/l/team/19%3a18b6456c0895433986694aaf4d6d0c56%40thread.tacv2/conversations?groupId=fc16c4db-e649-468c-b281-d739ce4198a2&tenantId=70a28522-969b-451f-bdb2-abfea3aaa5bf
Інформація про дисципліну	Дисципліна «Алгоритми та структури даних» є нормативною дисципліною з спеціальності 121 Інженерія програмного забезпечення для освітньої програми «Високопродуктивний комп'ютинг» яка викладається в 3 семестрі в обсязі 4,0 кредити (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено таким чином, щоб ознайомити учасників з класичними методами й задачами теорії алгоритмів, структурами даних а також найважливішими сучасними задачами інформатики. Вивчення цієї дисципліни має на меті зрозуміння та засвоєння основних принципів розробки алгоритмів і програм, а також дає підґрунтя для самостійної практичної роботи в галузі комп'ютерних наук. У курсі вивчаються інформаційні структури даних та обчислювальні моделі.
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни «Алгоритми та структури даних» є навчити студентів ефективно вирішувати алгоритмічні задачі, освоїти фундаментальні ідеї і методи теорії алгоритмів, виробити системний підхід до вирішення алгоритмічних задач, освоїти базові інформаційні структури даних, сформувати практичні навички розробки алгоритмів для розв'язання прикладних задач та їх програмування.
Література для вивчення дисципліни	<ol style="list-style-type: none"> 1. Клакович Л. М. Теорія алгоритмів : навчальний посібник. / Л. М. Клакович, С. М. Левицька, О. В. Костів. – Львів: ЛНУ, 2008. – 140 с. 2. T. H. Cormen, Ch. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms. Third Edition. – The MIT Press Cambridge, Massachusetts London, England. 3. A. V. Aho, J. E. Hopcroft, J. D. Ullman. Data structure and Algorithms. – Addison-Wesley. 4. R. Sedgewick. Algorithms in C++. Fundamentals Data Structures, Sorting, Searching, Graph Algorithms. – Addison-Wesley. 5. R. Sedgewick, K. Wayne. Algorithms. Fourth Editions. – Addison-Wesley. 6. Глибовець М.М. Основи комп'ютерних алгоритмів. – К.: Вид. дім „КМ академія”, 2003. – 452 с.

	<p>7. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 1. Fundamentals Algorithms. – Addison-Wesley.</p> <p>8. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 2. Seminumerical Algorithms. – Addison-Wesley.</p> <p>9. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 3. Sorting and Searching. – Addison-Wesley.</p> <p>10. J. J. McConnell. Analysis of Algorithms : An Active Learning Approach. Second Edition. – Jones and Bartlett Publishers.</p> <p>11. Нікольский Ю. В. Дискретна математика : Підручник. Видання третє, виправлене та доповнене / Ю. В. Нікольський, В. В. Пасічник, Ю. М. Щербина. – Львів: „Магнолія – 2006”, 2013. – 432 с.</p> <p>12. Шаховська Н. Б. Алгоритми і структури даних : посібник. / Н. Б. Шаховська, Р. О. Голощук. – Львів: „Магнолія – 2006”, 2010. – 215 с.</p>
Обсяг курсу	64 години аудиторних занять. З них 32 години лекцій, 32 години лабораторних робіт та 56 годин самостійної роботи
Очікувані результати навчання	<p>Після завершення цього курсу студент буде:</p> <ul style="list-style-type: none"> - Знати базові поняття теорії алгоритмів, способи представлення алгоритмів, основні алгоритмічні конструкції, принципи проектування алгоритмів, основні структури даних та операції над ними. - Вміти розробляти алгоритми та комп’ютерні програми мовами високого рівня для інформаційних систем; вибирати ефективну структуру даних для поставленої задачі; розробляти відповідно для структури даних алгоритм; використовувати рекурсивні структури даних та рекурсивні алгоритми. <p>Після вивчення даного курсу здобувачі набудуть таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК05. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ФК26. Здатність до алгоритмічного та логічного мислення.</p> <p>ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи різні методології та засоби програмування з метою забезпечення їх високої надійності та продуктивності в роботі.</p> <p>ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об’єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН26. Знати засоби інтеграції, розгортання та підтримки спеціалізованих програмних компонентів, розроблених на основі інноваційних технологій для вирішення завдань високопродуктивних обчислень.</p>

	ПРН27. Знати основи інженерії даних і конструювання конвеєрів даних та вміти обирати оптимальні алгоритми і технології розробки інноваційних рішень, зокрема для вирішення задач наук про дані та вбудованих систем.
Ключові слова	алгоритм, складність алгоритму, методи сортування, рекурсія, динамічні структури даних, бінарні дерева, хешування.
Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Іспит в кінці семестру
Пререквізити	Для вивчення курсу студенти потребують знань з дисциплін: „Дискретна математика”, „Вища математика”, „Програмування”.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	Мультимедіа, персональні комп'ютери з відповідним програмним забезпеченням, платформа MsTeams, платформа Moodle
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться упродовж семестру за 100-бальною шкалою. Бали нараховуються за такими видами робіт з наступним співвідношенням:</p> <ul style="list-style-type: none"> • лабораторні роботи: 30% семестрової оцінки; максимальна кількість балів 30. • контрольні заміри (2 модулі): 20% семестрової оцінки; максимальна кількість балів 20. • іспит: 50% семестрової оцінки; максимальна кількість балів 50. <p>Загалом упродовж семестру 100 балів.</p> <p>Контрольні заміри проводяться у формі тестових завдань.</p> <p>Академічна доброчесність: Очікується, що лабораторні та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями</p>

	під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.
Питання до контрольних робіт	Перелік питань та завдань для проведення підсумкової оцінки знань певних тем до контрольних робіт розміщені на веб-сторінці.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	Тема 1. Базові поняття про алгоритми. Вступ. Історичні аспекти. Формалізація поняття алгоритму. Алфавітні оператори та алгоритми. Способи задання алфавітних операторів. Властивості алгоритмів. Різновиди алгоритмів. Композиції алгоритмів.	Лекція	1, 2, 9, 10		
2	Тема 2. Аналіз алгоритмів. Сортування включенням. Аналіз алгоритму сортування методом включення. Порядок зростання. Асимптотичні позначення.	Лекція	2, 3, 4		
3	Тема 3. Метод декомпозиції. Метод декомпозиції („поділай та владарюй“). Сортування злиттям. Аналіз алгоритму сортування злиттям.	Лекція	2, 5, 7, 8, 10	Сортування методом злиття	3 тиж. семестру
4	Тема 4. Рекурентні співвідношення. Основні поняття. Метод підстановки. Метод дерев рекурсії. Основний метод.	Лекція	2, 4, 10		
5	Тема 5. Швидке сортування. Опис алгоритму швидкого сортування. Ефективність алгоритму швидкого сортування. Випадкова версія алгоритму швидкого сортування. Аналіз алгоритму швидкого сортування. Порядкові статистики (вибір за лінійний час).	Лекція	2, 5, 7, 8	Швидке сортування. Пошук порядкових статистик.	4 тиж. семестру 5 тиж. семестру
6	Тема 6. Сортування за лінійний час. Нижня оцінка алгоритмів сортування. Сортування підрахунком. Сортування за розрядами.	Лекція	2, 5	Сортування підрахунком. Сортування за розрядами.	6 тиж. семестру 7 тиж. семестру
7	Тема 7. Піраміди. Структура даних типу піраміда. Підтримка властивості піраміди. Створення піраміди. Алгоритм пірамідального сортування. Черги з пріоритетами.	Лекція	2, 5, 7, 10	Реалізація структури даних типу піраміда. Пірамідальне сортування. Реалізація черги з пріоритетами.	8 тиж. семестру 9 тиж. семестру
8	Тема 8. Хеш-таблиці. Таблиці з прямою адресацією. Хеш-таблиці. Уникнення колізій за допомогою ланцюжків	Лекція	2, 3, 5, 7, 8, 10	Побудова хеш-таблиці з відкритим хешуванням.	10 тиж. семестру

	(відкрите хешування). Хеш-функції. Відкрита адресація (закрите хешування).			Побудова хеш-таблиці з закритим хешуванням.	11 тиж. семестру
9	Тема 9. Бінарні дерева пошуку. Структура даних бінарне дерево пошуку. Робота з бінарними деревами пошуку. Операції вставки та видалення.	Лекція	2, 3, 5, 7, 8, 10	Реалізація структури даних типу бінарне дерево пошуку.	12 тиж. семестру
10	Тема 10. Деякі методи розробки ефективних алгоритмів. Методи розробки ефективних алгоритмів. Метод часткових цілей. Метод підйому. Метод відпрацювання назад.	Лекція	1, 2, 3, 6, 7, 8, 9, 10	Реалізація методу відпрацювання назад (бектрекінгу) для задачі про вісім ферзів.	13 тиж. семестру
11	Тема 10. Деякі методи розробки ефективних алгоритмів. Евристичні алгоритми. Метод гілок та меж. Динамічне програмування. Жадібні алгоритми.	Лекція	1, 2, 3, 6, 7, 8, 9, 10	Реалізація алгоритму динамічного програмування для задачі про перемноження матриць. Реалізація алгоритмів дискретної та неперервної задач про ранець. Реалізація методу гілок і меж для задачі комівояжера.	14 тиж. семестру 15 тиж. семестру 16 тиж. семестру
12	Тема 11 Алгоритмічні системи. Нормальні алгоритми. Поняття про алгоритмічні системи. Класичні алгоритмічні системи. Граф-схеми алгоритмів. Система нормальних алгоритмів Маркова. Принцип нормалізації.	Лекція	1, 2		
13	Тема 12. Обчислювальні функції. Рекурсивні функції. Зведення довільних алгоритмів до числових функцій. Обчислювальні функції. Найпростіші функції. Головні оператори. Примітивно-рекурсивні, частково-рекурсивні, загально-рекурсивні функції. Теза Черча.	Лекція	1, 2		
14	Тема 13. Машина Тюрінга. Алгоритмічна система Тюрінга. Формальне визначення машини Тюрінга. Теза Тюрінга. Різновиди машин Тюрінга. Алгоритмічно нерозв'язні задачі: проблема розпізнавання самозастосовності алгоритмів та проблема зупинки.	Лекція	1, 2		
15	Тема 14. Модель РАМ. Операторні алгоритмічні системи. Рівнодоступна адресна машина. Обчислювальна складність РАМ-програм. Зв'язок машин Тюрінга і РАМ.	Лекція	1, 2		
16	Тема 15. Класи складностей алгоритмів. Важко-розв'язні задачі. Недетерміновані алгоритми.	Лекція	1, 2		

	Класи P- та NP-повноти. Поліноміальна звідність, NP-повні задачі. Приклади NP-повних задач. NP-важкі задачі.				
--	--	--	--	--	--