

**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 № *2/22* від *30.08.2022* р.)

Завідувач кафедри:



Іван КАРБОВНИК

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

Львів 2022 р.

<b>Назва дисципліни</b>	Алгоритми і структури даних
<b>Адреса викладання дисципліни</b>	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова, 50, м. Львів, 79005, вул. ген. М. Тарнавського, 107, м. Львів, 79017
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет електроніки та комп'ютерних технологій Кафедра радіофізики та комп'ютерних технологій
<b>Галузь знань, шифр та назва спеціальності</b>	12 – інформаційні технології 121 – Інженерія програмного забезпечення
<b>Викладачі дисципліни</b>	Вельгош Сергій Романович, кандидат фізико-математичних наук, доцент, доцент кафедри радіофізики та комп'ютерних технологій
<b>Контактна інформація викладачів</b>	serhiy.velhosh@lnu.edu.ua <a href="https://electronics.lnu.edu.ua/employee/velhosh-s-r">https://electronics.lnu.edu.ua/employee/velhosh-s-r</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	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.
<b>Сторінка курсу</b>	<a href="https://teams.microsoft.com/l/team/19%3a18b6456c0895433986694aaf4d6d0c56%40thread.tacv2/conversations?groupId=fc16c4db-e649-468c-b281-d739ce4198a2&amp;tenantId=70a28522-969b-451f-bdb2-abfea3aaa5bf">https://teams.microsoft.com/l/team/19%3a18b6456c0895433986694aaf4d6d0c56%40thread.tacv2/conversations?groupId=fc16c4db-e649-468c-b281-d739ce4198a2&amp;tenantId=70a28522-969b-451f-bdb2-abfea3aaa5bf</a>  <a href="https://e-learning.lnu.edu.ua/course/view.php?id=278">https://e-learning.lnu.edu.ua/course/view.php?id=278</a>
<b>Інформація про дисципліну</b>	Дисципліна „Алгоритми і структури даних” є нормативною дисципліною зі спеціальності 121 Інженерія програмного забезпечення для освітньої програми „Інженерія програмного забезпечення” яка викладається в 3 семестрі в обсязі 3,5 кредити (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Коротка анотація дисципліни</b>	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 інженерії програмного забезпечення. У курсі вивчаються інформаційні структури даних та обчислювальні моделі.
<b>Мета та цілі дисципліни</b>	<i>Метою</i> вивчення нормативної дисципліни „Алгоритми і 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, сформувані практичні навички розробки алгоритмів для розв’язання прикладних задач та їх програмування. <i>Цілями</i> курсу є ознайомлення студентів з основними поняттями та проблемами, а також опанування фундаментальним для інформатики

	поняттями алгоритму, сформування практичних навичок розробки алгоритмів та вибору структур даних для розв'язання прикладних задач та їх програмування.
<b>Література для вивчення дисципліни</b>	<p style="text-align: center;">Основна література.</p> <ol style="list-style-type: none"> <li>1. Клакович Л. М. Теорія алгоритмів : навчальний посібник. / Л. М. Клакович, С. М. Левицька, О. В. Костів. – Львів: ЛНУ, 2008. – 140 с.</li> <li>2. T. H. Cormen, Ch. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms. Third Edition. – The MIT Press Cambridge, Massachusetts London, England.</li> <li>3. A. V. Aho, J. E. Hopcroft, J. D. Ullman. Data structure and Algorithms. – Addison-Wesley.</li> <li>4. R. Sedgewick. Algorithms in C++. Fundamentals Data Structures, Sorting, Searching, Graph Algorithms. – Addison-Wesley.</li> <li>5. R. Sedgewick, K. Wayne. Algorithms. Fourth Editions. – Addison-Wesley.</li> <li>6. Кренивч А. П. Алгоритми і структури даних : підручник. / А. П. Кренивч. – К. : ВПЦ „Київський університет”, 2021. – 200 с.</li> <li>7. Глибовець М.М. Основи комп'ютерних алгоритмів. – К.: Вид. дім „КМ академія”, 2003. – 452 с.</li> </ol> <p style="text-align: center;">Допоміжна література.</p> <ol style="list-style-type: none"> <li>8. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 1. Fundamentals Algorithms. – Addison-Wesley.</li> <li>9. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 2. Seminumerical Algorithms. – Addison-Wesley.</li> <li>10. D. E. Knuth. The Art of Computer Programming. Third Editions. – Vol. 3. Sorting and Searching. – Addison-Wesley.</li> <li>11. J. J. McConnell. Analysis of Algorithms : An Active Learning Approach. Second Edition. – Jones and Bartlett Publishers.</li> <li>12. Кублій Л. І. Алгоритми і структури даних. Основи алгоритмізації : підручник. / Л. І. Кублій. – К. : КПІ ім. Ігоря Сікорського, 2022. – 528 с.</li> <li>13. Нікольский Ю. В. Дискретна математика : Підручник. Видання третє, виправлене та доповнене / Ю. В. Нікольский, В. В. Пасічник, Ю. М. Щербина. – Львів: „Магнолія – 2006”, 2013. – 432 с.</li> <li>14. Шаховська Н. Б. Алгоритми і структури даних : посібник. / Н. Б. Шаховська, Р. О. Голощук. – Львів: „Магнолія – 2006”, 2010. – 215 с.</li> </ol>
<b>Обсяг курсу</b>	Загальний обсяг: 105 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 41 год.
<b>Очікувані результати навчання</b>	У результаті вивчення даного курсу студент буде: 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;

	<p>використовувати рекурсивні структури даних та рекурсивні алгоритми.</p> <p>Після вивчення даного курсу „Алгоритми і структури даних” здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.  ЗК2. Здатність застосовувати знання у практичних ситуаціях.  ЗК5. Здатність вчитися і оволодівати сучасними знаннями.  ЗК6. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.  ФК26. Здатність до алгоритмічного та логічного мислення.  ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи сучасні парадигми програмування.  ПРН5. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об’єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.  ПРН7. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.  ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.  ПРН26. Знати та вміти застосовувати засоби інженерії програмного забезпечення для реалізації проєктів з використанням технологій науки про дані та штучного інтелекту.  ПРН27. Вміти обирати оптимальні алгоритми та технології розробки програмного забезпечення.</p>
<b>Ключові слова</b>	Алгоритм, складність алгоритму, методи сортування, рекурсія, динамічні структури даних, бінарні дерева, хешування.
<b>Формат курсу</b>	Очний
<b>Теми</b>	<p>Тема 1. Базові поняття про алгоритми.  Тема 2. Аналіз алгоритмів.  Тема 3. Метод декомпозиції.  Тема 4. Швидке сортування.  Тема 5. Рекурентні співвідношення.  Тема 6. Сортування за лінійний час.  Тема 7. Бінарні дерева пошуку.  Тема 8. Структура даних типу піраміда.  Тема 9. Каркаси.  Тема 10. Хеш-таблиці.  Тема 11. Деякі методи розробки ефективних алгоритмів.  Тема 12. Деякі методи розробки ефективних алгоритмів (продовження).  Тема 13 Алгоритмічні системи. Нормальні алгоритми. Обчислювальні функції.  Тема 14. Машина Тюрінга.  Тема 15. Модель RAM.  Тема 16. Класи складностей алгоритмів.</p>
<b>Підсумковий</b>	Іспит у кінці семестру

<b>контроль, форма</b>																																																																					
<b>Пререквізити</b>	Для вивчення курсу студенти потребують знання з курсів: <ul style="list-style-type: none"> <li>• вища математика (1-ий, 2-ий семестри);</li> <li>• дискретна математика</li> <li>• основи програмування;</li> </ul>																																																																				
<b>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</b>	Лекції, презентації, виконання та захист лабораторних робіт, обговорення/консультації для кращого розуміння тем, самостійна робота, обговорення, дискусія.																																																																				
<b>Необхідне обладнання</b>	Мультимедійне обладнання (проектор), комп'ютерний клас, інтегральне середовище розробки (IDE), сервіси MS Teams, Moodle																																																																				
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> <li>• Змістовий модуль 1: 12 балів.</li> <li>• Змістовий модуль 2: 18 балів.</li> <li>• Змістовий модуль 3: 12 балів.</li> <li>• Змістовий модуль 4: 8 балів.</li> <li>• Модульний контроль у вигляді тестування у системі Moodle. Максимальна кількість балів за кожен тест – 10 балів. Модульний контроль проводиться після 2-го та 4-го змістових модулів.</li> <li>• Іспит з курсу. Максимальна кількість балів – 50 балів. Іспит складається з результатів модульних контрольних (максимальна кількість балів – 20), тесту в системі Moodle (максимальна кількість балів – 10), виконання практичного завдання по тематиці лабораторних робіт (максимальна кількість балів – 10), усної співбесіди з викладачем (максимальна кількість балів – 10).</li> </ul> <p>Підсумкова максимальна кількість балів 100.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th colspan="10">Поточне тестування та самостійна робота</th> </tr> <tr> <th colspan="5">Змістовий модуль 1</th> <th colspan="5">Змістовий модуль 2</th> </tr> <tr> <th>T1</th><th>T2</th><th>T3</th><th>T4</th><th>T5</th><th>T6</th><th>T7</th><th>T8</th><th>T9</th><th>T10</th> </tr> </thead> <tbody> <tr> <td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>5</td><td>4</td><td>4</td><td>5</td> </tr> </tbody> </table> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th colspan="6">Поточне тестування та самостійна робота</th> <th rowspan="2">Підсумковий іспит</th> <th rowspan="2">Сума</th> </tr> <tr> <th colspan="3">Змістовий модуль 3</th> <th colspan="3">Змістовий модуль 4</th> </tr> <tr> <th>T11</th><th>T12</th><th>T13</th><th>T14</th><th>T15</th><th>T16</th><td rowspan="2">50</td><td rowspan="2">100</td> </tr> </thead> <tbody> <tr> <td>6</td><td>6</td><td>2</td><td>2</td><td>2</td><td>2</td> </tr> </tbody> </table> <p><b>Академічна доброчесність:</b> 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>	Поточне тестування та самостійна робота										Змістовий модуль 1					Змістовий модуль 2					T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	2	2	2	2	2	2	5	4	4	5	Поточне тестування та самостійна робота						Підсумковий іспит	Сума	Змістовий модуль 3			Змістовий модуль 4			T11	T12	T13	T14	T15	T16	50	100	6	6	2	2	2	2
Поточне тестування та самостійна робота																																																																					
Змістовий модуль 1					Змістовий модуль 2																																																																
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10																																																												
2	2	2	2	2	2	5	4	4	5																																																												
Поточне тестування та самостійна робота						Підсумковий іспит	Сума																																																														
Змістовий модуль 3			Змістовий модуль 4																																																																		
T11	T12	T13	T14	T15	T16	50	100																																																														
6	6	2	2	2	2																																																																

	<p><b>Відвідання занять</b> 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><b>Література.</b> 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><b>Політика виставлення балів.</b> 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 під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</p>
<p><b>Питання до заліку чи екзамену.</b></p>	<ol style="list-style-type: none"> <li>1. Часова, ємнісна складність алгоритмів. Поліноміальна, експоненційна складність. Асимптотичні позначення (<math>\Theta</math>, <math>\Omega</math>, <math>O</math>) та їхні властивості.</li> <li>2. Основна теорема (метод) розв'язку рекурентних співвідношень.</li> <li>3. Абстрактні алфавіти. Операції конкатенації, входження, заміни. Алфавітний оператор. Кодувальні алфавітні оператори. Способи задання алфавітних операторів.</li> <li>4. Властивості алгоритмів: дискретність, ефективність, скінченність, результативність, масовість. Різновиди алгоритмів: детерміновані (недетерміновані), самозмінні (несамозмінні), самозастосовні (несамозастосовні), універсальні.</li> <li>5. Композиції алгоритмів: суперпозиція, об'єднання, розгалуження, ітерація.</li> <li>6. Нормальні алгоритми Маркова: оператор підстановки і розпізнавач входження; принцип роботи нормального алгоритму; принцип нормалізації.</li> <li>7. Рекурсивні функції. Поняття обчислювальної функції; найпростіші функції: функція наступності, нуль-функція, функція вибору аргументів. Оператор суперпозиції, оператор примітивної рекурсії, оператор мінімізації. Примітивно-рекурсивні функції. Частково-рекурсивні функції. Загально-рекурсивні функції. Теза Черча.</li> <li>8. Машина Тьюрінга: будова, принцип роботи, програма. Теза Тьюрінга. Різновиди машини Тьюрінга. Проблема розпізнавання самозастосовності алгоритмів та проблема зупинки.</li> <li>9. Рівнодоступна адресна машина: будова, принцип роботи, операнди, команди.</li> <li>10. Поняття про важкорозв'язні задачі. Детерміновані та недетерміновані алгоритми.</li> <li>11. Класи складності P та NP. Проблема P = NP. Поліноміальна звідність. NP-повні задачі. Доведення NP-повноти задачі. Приклади NP-повних задач. NP-важкі задачі.</li> <li>12. Метод декомпозиції. Сортування злиттям. Аналіз сортування злиттям.</li> <li>13. Швидке сортування. Аналіз швидкого сортування. Порядкові статистики.</li> </ol>

	<p>14. Сортування за лінійний час. Сортування підрахунком та за розрядами. Стійкість алгоритмів сортування.</p> <p>15. Піраміди. Побудова піраміди. Підтримка властивості піраміди. Пірамідальне сортування. Черги з пріоритетами.</p> <p>16. Бінарні дерева пошуку. Операції вставки, пошуку, видалення у бінарному дереві пошуку.</p> <p>17. Каркаси. Алгоритми Прима та Крускала для побудови каркасів мінімальної ваги.</p> <p>18. Хеш-таблиці. Таблиці з прямою адресацією. Принцип хешування.</p> <p>19. Уникнення колізій за допомогою ланцюгів (відкрите хешування). Аналіз методу ланцюгів.</p> <p>20. Хеш-функції: метод ділення, метод множення, їх аналіз.</p> <p>21. Відкрита адресація (закрите хешування): вставка, пошук, видалення елементів. Побудова хеш-функції при закритому хешуванні: лінійне дослідження, квадратичне дослідження, подвійне хешування. Аналіз методу відкритої адресації.</p> <p>22. Жадібні алгоритми.</p> <p>23. Метод відпрацьовування назад. Задача про вісім ферзів.</p> <p>24. Динамічне програмування. Задача про перемноження матриць.</p> <p>25. Евристичні алгоритми.</p> <p>26. Метод гілок та меж. Задача комівояжера.</p>
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

**Схема курсу „Алгоритми і структури даних”  
для студентів спеціальності 121 – Інженерія програмного забезпечення**

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.* ** Ресурси в Інтернеті	Завдання, год	Термін виконання
1	<b>Тема 1. Базові поняття про алгоритми.</b> Вступ. Історичні аспекти. Формалізація поняття алгоритму. Алфавітні оператори та алгоритми. Способи задання алфавітних операторів. Властивості алгоритмів. Різновиди алгоритмів. Композиції алгоритмів.	Лекція	[1], [2], [3], [4], [8], [11] Сайт курсу	2	
1	Вступне заняття.	Лабораторне заняття	Сайт курсу	2	
2	<b>Тема 2. Аналіз алгоритмів.</b> Сортування включенням. Аналіз алгоритму сортування методом включення. Порядок зростання. Асимптотичні позначення.	Лекція	[2], [3], [4], [5], [6], [8], [11] Сайт курсу	2	
2	Сортування методом включення.	Лабораторне заняття	Сайт курсу	2	3 тиждень семестру
3	<b>Тема 3. Метод декомпозиції.</b> Метод декомпозиції („поділяй та владарюй”). Сортування злиттям. Аналіз алгоритму сортування злиттям.	Лекція	[2], [4], [5], [6], [7], [10], [11], [14] Сайт курсу	2	
3	Сортування методом злиття.	Лабораторне заняття	Сайт курсу	2	4 тиждень семестру
4	<b>Тема 4. Швидке сортування.</b> 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], [3], [4], [5], [6], [10], [11], [14] Сайт курсу	2	
4	Швидке сортування.	Лабораторне заняття	Сайт курсу	2	5 тиждень семестру
5	<b>Тема 5. Рекурентні співвідношення.</b> Основні поняття. Метод підстановки. Метод дерев рекурсії. Основний метод.	Лекція	[2], [3], [11] Сайт курсу	2	
5	Пошук порядкових статистик.	Лабораторне заняття	Сайт курсу	2	6 тиждень семестру
6	<b>Тема 6. Сортування за лінійний час.</b> Нижня оцінка алгоритмів сортування, що базуються на порівнянні елементів. Сортування підрахунком. Сортування за розрядами.	Лекція	[2], [10], [11], [14] Сайт курсу	2	
6	Сортування підрахунком. Сортування за розрядами.	Лабораторне заняття	Сайт курсу	2	7 тиждень семестру
7	<b>Тема 7. Бінарні дерева пошуку.</b> Структура даних бінарне дерево пошуку. Робота з бінарними деревами	Лекція	[2], [3], [4], [5], [6], [8], [14]	2	



	пошуку. Операції пошуку, вставки та видалення.		Сайт курсу		
7	Реалізація структури даних типу бінарне дерево пошуку.	Лабораторне заняття	Сайт курсу	2	8 тиждень семестру
8	<b>Тема 8. Структура даних типу піраміда.</b> Структура даних типу піраміда. Підтримка властивості піраміди. Створення піраміди. Алгоритм пірамідального сортування. Черги з пріоритетами.	Лекція	[2], [3], [4], [5] Сайт курсу	2	
8	Реалізація операцій пошуку, вставки та видалення і бінарному дереві пошуку.	Лабораторне заняття	Сайт курсу	2	9 тиждень семестру
9	<b>Тема 9. Каркаси.</b> Каркаси або з'єднувальні дерева. Цикломатичне число. Каркас мінімальної ваги. Алгоритм Краскала пошуку каркасу мінімальної ваги. Алгоритм Пріма пошуку каркасу мінімальної ваги.	Лекція	[2], [4], [5], [11], [13] Сайт курсу	2	
9	Реалізація структури даних типу піраміда. Пірамідальне сортування.	Лабораторне заняття	Сайт курсу	2	10 тиждень семестру
10	<b>Тема 10. Хеш-таблиці.</b> Таблиці з прямою адресацією. Хеш-таблиці. Уникнення колізій за допомогою ланцюжків (відкрите хешування). Хеш-функції. Відкрита адресація (закрите хешування).	Лекція	[2], [4], [5], [6], [14] Сайт курсу	2	
10	Реалізація черги з пріоритетами на основі піраміди.	Лабораторне заняття	Сайт курсу	2	11 тиждень семестру
11	<b>Тема 11. Деякі методи розробки ефективних алгоритмів.</b> Методи розробки ефективних алгоритмів. Метод часткових цілей. Метод підйому. Метод відпрацьовування назад.	Лекція	[1], [3], [6], [7], [11] Сайт курсу	2	
11	Побудова каркасу мінімальної ваги методами Краскала та Пріма.	Лабораторне заняття	Сайт курсу	2	12 тиждень семестру
12	<b>Тема 12. Деякі методи розробки ефективних алгоритмів (продовження).</b> Жадібні алгоритми. Динамічне програмування. Евристичні алгоритми. Метод гілок та меж.	Лекція	[1], [2], [3], [6], [7], [11], [14] Сайт курсу	2	
12	Побудова хеш-таблиці з відкритим хешуванням.	Лабораторне заняття	Сайт курсу	2	13 тиждень семестру
13	<b>Тема 13 Алгоритмічні системи. Нормальні алгоритми. Обчислювальні функції.</b> Поняття про алгоритмічні системи. Класичні алгоритмічні системи. Граф-схеми алгоритмів. Система нормальних алгоритмів Маркова. Принцип нормалізації. 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], [13], [14] Сайт курсу	2	

13	Побудова хеш-таблиці з закритим хешуванням.	Лабораторне заняття	Сайт курсу	2	14 тиждень семестру
14	<b>Тема 14. Машина Тюрінга.</b> 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], [13], [14] Сайт курсу	2	
14	Реалізація алгоритму динамічного програмування для задачі про перемноження матриць.	Лабораторне заняття	Сайт курсу	2	15 тиждень семестру
15	<b>Тема 15. Модель RAM.</b> Операторні алгоритмічні системи. Рівнодоступна адресна машина. Обчислювальна складність RAM-програм. Зв'язок машин Тюрінга і RAM.	Лекція	[1], [7], [14] Сайт курсу	2	
15	Реалізація методу відпрацьовування назад (бектрекінгу) для задачі про вісім ферзів.	Лабораторне заняття	Сайт курсу	2	16 тиждень семестру
16	<b>Тема 16. Класи складностей алгоритмів.</b> Важко-розв'язні задачі. Недетерміновані алгоритми. Класи P- та NP-повноти. Поліноміальна звідність, NP-повні задачі. Приклади NP-повних задач. NP-важкі задачі.	Лекція	[1], [2], [7], [11], [13] Сайт курсу	2	
16	Заключне лабораторне заняття.	Лабораторне заняття	Сайт курсу	2	

<https://teams.microsoft.com/l/team/19%3a18b6456c0895433986694aaf4d6d0c56%40thread.tacv2/conversations?groupId=fc16c4db-e649-468c-b281-d739ce4198a2&tenantId=70a28522-969b-451f-bdb2-abfea3aaa5bf>

<https://e-learning.lnu.edu.ua/course/view.php?id=278>