



Національний технічний університет
України "Київський політехнічний
інститут імені Ігоря Сікорського"



Інститут спеціального зв'язку та захисту
інформації КПІ ім. Ігоря Сікорського
Спеціальна кафедра № 5

ПРОЕКТУВАННЯ ТА АНАЛІЗ ОБЧИСЛЮВАЛЬНИХ АЛГОРИТМІВ

Робоча програма навчальної дисципліни (силабус)

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Освітньо-професійна програма	<i>Комп'ютерні системи і технології спеціального зв'язку</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (Денна)</i>
Рік підготовки, семестр	<i>II рік підготовки, весняний семестр</i>
Обсяг дисципліни	<i>3 кредити ECTS</i>
Семестровий контроль / контрольні заходи	<i>Залік</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лекції: Дмитро ШАРАДКІН Практичні: Ольга ШЕВЧУК</i>
Розміщення курсу	<i>Google Classroom</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Силабус освітнього компонента «Проектування та аналіз обчислювальних алгоритмів» складено відповідно до освітньої програми підготовки бакалаврів «Комп'ютерні системи і технології спеціального зв'язку» спеціальності 122 – Комп'ютерні науки.

Метою навчальної дисципліни є формування та закріплення у курсантів наступних компетентностей: (ЗК 1) Здатність до абстрактного мислення, аналізу та синтезу; (ЗК 2) Здатність застосовувати знання у практичних ситуаціях; (ЗК 3) Знання та розуміння предметної області та розуміння професійної діяльності; (ЗК 6) Здатність вчитися й оволодівати сучасними знаннями; (ЗК 7) Здатність до пошуку, оброблення та аналізу інформації з різних джерел; (ЗК 8) Здатність генерувати нові ідеї (креативність); (ЗК 11) Здатність приймати обґрунтовані рішення; (СК 1) Здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування; (СК 7) Здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів; (СК 8) Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління; (СК 15) Здатність до аналізу та функціонального моделювання бізнес-процесів, побудови та практичного застосування функціональних моделей організаційно-економічних і виробничо-технічних систем, методів оцінювання ризиків їх проектування.

Предмет навчальної дисципліни - методи, алгоритми проектування та аналізування обчислювальних алгоритмів.

Програмні результати навчання, на формування та покращення яких спрямована дисципліна: (ПР 1) Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук; (ПР 2) Використовувати сучасний математичний апарат неперервного та дискретного аналізу, лінійної алгебри, аналітичної геометрії, в професійній діяльності для розв'язання задач теоретичного та прикладного характеру в процесі проектування та реалізації об'єктів інформатизації; (ПР 3) Використовувати знання закономірностей випадкових явищ, їх властивостей та операцій над ними, моделей випадкових процесів та сучасних програмних середовищ для розв'язування задач статистичної обробки даних і побудови прогнозних моделей; (ПР 5) Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

2. навчальної дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Для успішного засвоєння дисципліни курсант повинен володіти освітніми компонентами «Дискретна математика», «Теорія ймовірності, ймовірнісні процеси і математична статистика» та «Математичний аналіз». Компетенції, знання та уміння, одержані в процесі вивчення освітнього компонента є необхідними для подальшого вивчення освітніх компонентів «Системний аналіз» та «Теорія прийняття рішень».

3. Зміст навчальної дисципліни**Семестр 4****Семестровий (кредитний) модуль 1.****Тема 1. Програмні засоби проектування обчислювальних алгоритмів.**

Основи мови програмування Python. Використання мови програмування Python.
Практика програмування на Python.

Тема 2. Алгоритми сортування.

Основи теорії алгоритмів. Поняття алгоритму. Поняття складності алгоритму.
Методи сортування. Швидке та пірамідальне сортування. Спеціальні способи сортування та обчислення порядкових статистик. Складання та налаштування програм сортування вибором на мові програмування Python.

Тема 3. Базові структури даних та робота з ними.

Поняття абстрактної структури даних та її реалізації. Множини, стеки та черги, зв'язані списки та дерева. Бінарні дерева пошуку. Реалізація базових структур даних та алгоритми роботи з ними.

Тема 4. Обчислювальні алгоритми на графах.

Графи та їх подання на комп'ютері. Алгоритми пошуку шляхів в графі. Пошук в ширину та глибину. Представлення графів в програмах та робота з ними.

4. Навчальні матеріали та ресурси**Основна література:**

1. *Aho, A. V.* The Design and Analysis of Computer Algorithms. / Aho, A. V., Hopcroft, J. E., Ullman, J. D. – London-Amsterdam-Don Mills-Sydney. Addison-Wesley Publ. Comp. 1979. – 479 p.
2. *Wirth.N* Algorithms and Data Structures / N. Wirth. – Oberon version: August 2004. – 212 p.
3. *Dasgupta S.* Algorithms./ S Dasgupta, C Papadimitriou, U Vazirani. – McGraw-Hill Science/Engineering/Math; 1 edition, 2006. – 336p.
4. *Corman T.* Algorithms. Construction and analysis. / Thomas H. Corman, Charles E.Lazerson. – The MIT Press; 3rd edition, 2009. – 1292p.
5. *Коротеева Т.О.* Алгоритми та структури даних.-Львів, Видавництво львівської політехніки, 2014.-280с.
6. *Ільман В.* Алгоритми, дані і структури. Навч. посіб. / В.М. Ільман, О.П. Іванов, Л.О. Панік. Дніпропет. нац. ун-т залізн. трансп.ім. акад. В. Лазаряна. – Дніпро, 2019. – 134 с.
7. *Висоцька В.А.* Алгоритмізація та програмування. Навч. посіб. / В.А. Висоцька, О.В. Оборська, Міністерство освіти і науки України, Національний університет "Львівська політехніка" .. – Львів :Видавництво "Новий Світ-2000",2021.. – 514 с.
8. *Lutz M.* Learning Python, Fourth Edition. / M.Lutz – O`Reilly Media,Inc , 2009. –1280с.
9. *Ковалюк Т.В.* Алгоритмізація та програмування. / Т.В. Ковалюк —Львів :Видавництво "Магнолія 2006",2021. — 399 с.
10. *Крєневич А.П.* Python у прикладах і задачах. Частина 1. Структурне програмування
Навчальний посібник із дисципліни "Інформатика та програмування" / А.П. Крєневич– К.: ВПЦ "Київський Університет", 2017. – 206 с.
11. *Aditya Bargava.* Grokking Algorithms / Bagrava Aditya – Manning Publications 1st édition, 2016.-256с.

Додаткова література:

1. *Green G.* Mathematics for the analysis of algorithms / D. Green, D. Knuth. – Birkhauser Boston Inc, 2007. – 140 p.

2. *Knuth D.* The art of Computer Programming: Volume 2: Seminumerical Algorithms. / D. Knuth. – Addison-Wesley Professional; 3 edition, 2014. – 784 p.
3. *Knuth D.* The art of Computer Programming: Volume 3: Sorting and Searching. / D. Knuth. – Addison-Wesley Professional; 2 edition, 1998. – 800 p.
4. *Cormen T.* Introduction to Algorithms. / T. Cormen. – McGraw-Hill Science/Engineering/Math; 2nd edition, 2001. – 1056 p.
5. *Levitin A.* Introduction to the Design and Analysis of Algorithms. / A. Levitin. – Addison-Wesley, 2005. – 592 p.
6. *Reingold E.* Combinatorial Algorithms: Theory and Practice / E. Reingold, N. Deo. – Pearson College Div, 1977. – 930 p.
7. *Heineman G.* Algorithms in a Nutshell. / G. Heineman, — O'Reilly, 2008. — 360 p.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Структура кредитного модуля

Номери, назви розділів, тем і питання навчальних занять, посилання на літературу		Кількість годин				
		Всього	у тому числі			
			Лекції	Практичні (семінарські) заняття	Лабораторні заняття (комп'ютерний практикум)	СР
Тема 1	Програмні засоби проектування обчислювальних алгоритмів	15	4	6		5
Заняття 1/1	Основи мови програмування Python 1. Місце мови програмування Python серед інших мов програмування. 2. Особливості мови Python. 3. Базові та складові типи даних мови Python. 4. Основні конструкції мови Python. Основна література: [8,9,10,11]	3	2			1
Заняття 1/2	Використання мови програмування Python 1. Функції в мові Python. 2. Базові конструкції об'єктно-орієнтованого програмування. 3. Ввод-вивод інформації засобами Python. Основна література: [8,9,10,11]	3	2			1
Заняття 1/3	Встановлення, запуск та початок роботи з інтерпретатором Python 1. Інсталяція та запуск Python. 2. Інтерактивне середовище програмування. Його встановлення та робота. 3. Написання простих програм на Python. Основна література: [8,11]	3		2		1
Заняття 1/4	Практика програмування на Python. 1. Типи даних та оператори.	3		2		1

	2. Керуючі конструкції та їх використання. 3. Особливості використання функцій вводу-виводу в мові програмування Python. Основна література: [8,9,10,11]					
Заняття 1/5	Використання функцій та класів при програмуванні на Python. 1. Особливості використання функцій в Python. 2. Особливості вводу-виводу в мові програмування Python. 3. Бібліотека для наукових обчислень NumPy. Основна література: [8,9,10,11]	3		2		1
Тема 2	Алгоритми сортування	32	8	10		14
Заняття 2/1	Основи теорії алгоритмів 1. Поняття алгоритму. 2. Поняття складності алгоритму. 3. Види алгоритмів. Основна література: [1,2,3,4,5,6,7]	3	2			1
Заняття 2/2	Методи сортування 1. Метод сортування бульбашкою. 2. Метод сортування вставкою. 3. Метод сортування вибором. 4. Метод сортування злиттям. 5. Оцінювання обчислювальної складності методів сортування. Основна література: [1,2,3,4,5,6,7]	3	2			1
Заняття 2/3	Швидке та пірамідальне сортування 1. Метод швидкого сортування Хоара, оцінка його обчислювальної складності. 2. Структура даних типу піраміди. 3. Сортування на основі пірамід. Основна література: [1,2,3,4,5,6,7]	4	2			2
Заняття 2/4	Спеціальні способи сортування та обчислення порядкових статистик. 1. Сортування підрахунком. 2. Порозрядне сортування. 3. Сортування при рівномірному законі розміщення значень елементів. 4. Алгоритми пошуку мінімуму, максимуму та медіани серед елементів масиву. 5. Алгоритм пошуку і-го в порядку зростання елементу. Основна література: [1,2,3,4,5,6,7]	4	2			2

Заняття 2/5	Складання та налаштування програм сортування бульбашкою та вставкою на мові програмування Python. 1. Розробка та налаштування програм сортування бульбашкою та її модифікації. 2. Розробка та налаштування програм сортування вставкою. Основна література: [5,7,10]	3		2		1
Заняття 2/6	Складання та налаштування програм сортування злиттям на мові програмування Python. 1. Розробка та налаштування програм сортування злиттям. 2. Вивчення засобів визначення часу виконання програм на мові Python. Основна література: [5,7,10]	3		2		1
Заняття 2/7	Складання та налаштування програм сортування вибором на мові програмування Python. 1. Розробка та налаштування програм сортування злиттям. 2. Порівняння використання різних алгоритмів сортування з точки зору часу виконання програм. Основна література: [5,7,10]	3		2		1
Заняття 2/8	Програмна реалізація швидкого сортування Хоара. 1. Набуття практичних навиків в програмуванні алгоритму сортування Хоара та його різновидів Основна література: [5,7,10]	4		2		2
Заняття 2/9	Програмна реалізація пірамідального алгоритму сортування. 1. Набуття практичних навичок в написанні та налагодженні програм пірамідального сортування. 2. Контрольна робота. Основна література: [5,7,10]	5		2		3
Тема 3	Базові структури даних та робота з ними	21	4	8		9
Заняття 3/1	Структури даних. 1. Поняття абстрактної структури даних та її реалізації. 2. Динамічні множини. 3. Стеки та черги. 4. Зв'язані списки та кореневі дерева. Основна література: [1,2,3,4,5,6,7]	3	2			1
Заняття 3/2	Бінарні дерева пошуку.	3	2			1

	<p>1. Поняття бінарного дерева пошуку.</p> <p>2. Робота з бінарним деревом пошуку.</p> <p>3. Різновиди бінарних дерев пошуку.</p> <p>Основна література: [1,2,3,4,5,6,7]</p>					
Заняття 3/3	<p>Операції над зв'язаними списками.</p> <p>1. Практика складання та налаштування програм для роботи з зв'язаними списками – пошук, вставка, видалення елементів.</p> <p>2. Різновиди реалізації списків.</p> <p>Основна література: [5,7,10]</p>	4		2		2
Заняття 3/4	<p>Операції над стеками та чергами,</p> <p>1. Практика складання та налаштування програм для роботи з стеками та чергами.</p> <p>2. Операції над кореневими деревами.</p> <p>Основна література: [5,7,10]</p>	4		2		2
Заняття 3/5	<p>Дерева, та робота з ними реалізація структур типу «дерева» за допомогою списків.</p> <p>1. Практика складання та налаштування програм для роботи деревами.</p> <p>Основна література: [5,7,10]</p>	4		2		2
Заняття 3/6	<p>Реалізація операцій з бінарним деревом на мові програмування Python</p> <p>1. Алгоритм повного обходу дерева</p> <p>2. Алгоритм пошуку елемента з заданим ключем.</p> <p>3. Пошук мінімуму та максимуму в дереві.</p> <p>Основна література: [5,7,10]</p>	3		2		1
Тема 4	Обчислювальні алгоритми на графах	14	4	4		6
Заняття 4/1	<p>Графи та їх подання на комп'ютері.</p> <p>1. Означення, різновиди, властивості графів.</p> <p>2. Операції над графами.</p> <p>3. Представлення графів.</p> <p>4. Пошук на графах.</p> <p>Основна література: [1,2,3,4,5,6,7]</p>	3	2			1
Заняття 4/2	<p>Алгоритми пошуку шляхів в графі. Пошук в ширину та глибину.</p> <p>1. Постановка задачі.</p>	3	2			1

	2. Обхід графу в ширину та глибину. 3. Побудова найкоротших шляхів в графах. Алгоритм Дейкстри. Основна література: [1,2,3,4,5,6,7]				
Заняття 4/3	Представлення графів в програмах та робота з ними. 1. Способи представлення графів 2. Переходи між способами задання графів. 3. Реалізація операцій додавання та видалення елементів графу. Основна література: [5,7,10]	4		2	2
Заняття 4/4	Програмна реалізація алгоритму пошуку шляхів в графі. 1. Набуття навиків в написанні та налаштуванні алгоритмів обходу графа в ширину та глибину. 2. Реалізація алгоритму пошуку найкоротшого шляху в графі. Основна література: [5,7,10]	4		2	2
Залік		8		2	6
Всього годин		90	20	30	40

6. Самостійна робота курсанта

Головними видами самостійної роботи курсантів є: самостійна підготовка до аудиторних занять, підготовка домашніх завдань та самостійна підготовка до заліку.

Доцільно час самостійної підготовки для поглибленого вивчення та закріплення навчального матеріалу розподілити наступним чином:

№ з/п	Назва теми та перелік основних питань (перелік дидактичного забезпечення, посилання на літературу)	Кількість годин СР
1	Тема 1. Програмні засоби проектування обчислювальних алгоритмів. 1. Самостійно встановити інтерпретатор Python 2. Самостійно встановити інтегроване середовища для роботи з інтерпретатором Python 3. Ознайомитися з засобами налаштування програм. Основна література: [8,11]	5
2	Тема 2. Алгоритми сортування 1. Обґрунтування правильності алгоритмів 2. Форми представлення алгоритмів. 3. Індукція та рекурсія. 4. Випадки оцінювання обчислювальної складності. 5. Оцінювання складності при найгіршому та найкращому випадках. 6. Налаштування програм на Python, що реалізують алгоритми сортування Основна література: [1,4,7]	14
3	Тема 3. Базові структури даних та робота з ними 1. Елементарні структури даних. 2. Суміжні та зв'язні структури даних.	9

№ з/п	Назва теми та перелік основних питань (перелік дидактичного забезпечення, посилання на літературу)	Кількість годин СР
	3. Реалізація покажчиків та об'єктів. 4. Хешування. Хеш-функції. Хеш-таблиці. 5. Дерева цифрового пошуку. Основна література: [5,6,7,9]	
4	Тема 4. Обчислювальні алгоритми на графах 1. Перехід між різними формами представлення графу 2. Пошук найкоротших шляхів між усіма парами вершин. Основна література: [2,4,10]	6
5	Підготовка до заліку	6
	Всього годин	40

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

У ході навчальних занять використовуються наступні методи навчання:

- усне викладання матеріалу;
- обговорення учбового матеріалу;
- практична робота в класі з застосуванням комп'ютерної техніки;
- самостійна робота під керівництвом викладача.

Відвідування занять є обов'язковим. Відсутність на заняттях з будь-яких причин не вважається поважною причиною невиконання відповідного домашнього завдання.

Під час занять всі мобільні телефони мають бути переведені на беззвучний режим роботи. Під час занять заборонено надсилання текстових повідомлень, прослуховування музики, перевірка електронної пошти, соціальних мереж тощо. Електронні пристрої, включаючи мобільні телефони та ноутбуки можна використовувати лише за умови виробничої необхідності в них (за погодженням з викладачем).

Всі робочі оголошення та необхідні матеріали курсу будуть розміщуватися на відповідній сторінці в Google Class. Очікується, що студенти перевірятимуть свою електронну пошту і сторінку навчальної дисципліни в Google Class та реагуватимуть своєчасно. Результат виконання домашніх завдань також мають бути викладені на сторінці Google Class у форматі, який буде вказаний викладачем. Також через сторінку Google Class курсанти можуть надіслати у вигляді відкритого чи приватного листа викладачу питання, що виникли під час виконання завдань, або інші питання стосовно курсу, який вивчається.

Домашні завдання мають бути виконані і надіслані на перевірку виключно до дати, яка вказана як кінцевий термін її виконання. Завдання надіслані після вказаного строку можуть але не зобов'язані бути перевірені та оцінені викладачем.

Кожний студент зобов'язаний дотримуватися принципів академічної доброчесності. Письмові завдання з використанням часткових або повнотекстових запозичень з інших робіт без зазначення авторства – це плагіат. Використання будь-якої інформації (текст, фото, ілюстрації тощо) мають бути правильно процитовані з посиланням на автора. До студентів, у роботах яких буде виявлено списування, плагіат чи інші прояви недоброчесної поведінки можуть бути застосовані різні дисциплінарні заходи.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Видами контролю якості навчання здобувачів є: поточний, календарний та семестровий контроль.

Оцінювання результатів навчання курсантів здійснюється у відповідності до Методичних рекомендацій до розроблення і застосування рейтингових систем оцінювання курсантів (студентів) в ІСЗІ КПІ ім. Ігоря Сікорського.

Рейтинг курсанта з навчальної дисципліни складається з балів, що він отримує за:

- поточне опитування на лекційних та практичних заняттях,
- практична робота на комп'ютері в ході практичних занять;
- виконання домашніх практичних завдань;
- поточну контрольну роботу;
- підсумковий контроль.

Поточне опитування на лекційних та практичних заняттях.

Відповіді в ході опитування на лекційних та практичних заняттях оцінюються виходячи з їх повноти, системності, впевненості та кількості. Максимальна кількість балів за усні опитування на лекційних та практичних заняттях дорівнює 24 балам.

Критерії оцінювання:

1 – всі надані відповіді правильні та повні;

0.6-надані відповіді правильні, але не повні, містять певні несуттєві недоліки;

0 - в противному разі.

Таким чином, максимальна кількість балів, що її курсант може отримати по цьому виду контролю за семестр становить $1*10(\text{лекцій})+1*14(\text{пр.зан.})=24$ бали.

Практична робота на комп'ютері в ході практичних занять.

За виконання практичних завдань на комп'ютері курсанти отримують рейтингові бали виходячи з якості розробленого коду програми.

Критерії оцінювання:

2 – завдання виконано без зауважень;

1.2 – допущена неточність несуттєвого характеру, або при правильному ході рішення практично виконане завдання не закінчено;

0 – при рішенні допущено суттєві помилки;

Таким чином, максимальна кількість балів, що її курсант може отримати по цьому виду контролю за семестр становить $2*14=28$ балів.

Виконання домашніх практичних завдань.

За виконання домашніх практичних завдань курсанти отримують рейтингові бали виходячи з якості розробленого коду програми.

Критерії оцінювання:

3 – завдання виконано без зауважень;

2.4 – допущена неточність несуттєвого характеру;

1.8 – допущена неточність, або при правильному ході рішення практично виконане завдання не закінчено;

0 – при рішенні допущено суттєві помилки;

Таким чином, максимальна кількість балів, що її курсант може отримати по цьому виду контролю за семестр становить $3*14=42$ балів.

Поточна контрольна робота.

Поточна контрольна робота виконується після вивчення Теми 2 курсу, проводиться в формі написання комплексної задачі по сортуванню даних в використанні мови програмування Python. Рейтингові бали нараховуються виходячи з якості розробленого коду програми.

Критерії оцінювання:

6 – завдання виконано без зауважень;

5 – допущена неточність несуттєвого характеру;

4 – допущена неточність, або при правильному ході рішення практично виконане завдання не закінчено;

0-3 – при рішенні допущено суттєві помилки;

Максимальна кількість балів, що курсант може отримати по цьому виду контролю за семестр становить 6 балів.

Таким чином, максимальна кількість рейтингових балів, що курсант може отримати в семестрі по результатам поточного контролю та виконанню контрольних робіт складає:

$$RD=24 + 28 + 42 + 6 = 100 \text{ балів.}$$

Умовою атестації (календарного контролю) є отримання не менше 50% від кількості балів, яку курсант може отримати на час проведення атестації.

Умовою допуску до заліку є: виконання усіх видів робіт та завдань (практичні роботи, контрольна робота), що передбачені навчальним планом.

Підсумковий контроль з навчальної дисципліни провадиться в вигляді заліку, що проводиться наприкінці семестру.

Якщо сума балів менша за 60, курсант виконує залікову контрольну роботу.

Курсант, який набрав протягом семестру необхідну кількість балів більше 60, отримують залікову оцінку (залік) так званим «автоматом» відповідно до набраного рейтингу. В такому разі до заліково-екзаменаційної відомості вносяться бали та відповідні оцінки.

Курсант, який у семестрі отримав більше 60 балів, може взяти участь у заліковій контрольній роботі з метою підвищення оцінки. У цьому разі бали, отримані ним на заліковій контрольній роботі, є остаточними. Якщо оцінка за залікову роботу більша ніж за рейтингом, курсант отримує оцінку за результатами залікової контрольної роботи.

Якщо оцінка за залікову контрольну роботу менша, ніж за рейтингом попередній рейтинг курсанта з кредитного модуля скасовується і він отримує оцінку з урахуванням результатів залікової контрольної роботи

Залікова робота оцінюється з 100 балів. За виконання залікової роботи курсант може отримати наступні рейтингові бали:

95-100 балів – робота виконана повністю, в заданому обсязі. Робота виконувалась самостійно, всі положення роботи приведені аргументовано та впевнено, показані коректність зроблених висновків. Курсант використав найоптимальніший з можливих методів вирішення задачі, продемонстрував навички використання сучасних парадигм програмування.

85-94 балів – робота виконана повністю, в заданому обсязі. Робота виконувалась самостійно, всі положення роботи приведені аргументовано та впевнено, показані коректність зроблених висновків.

75-84 бали – робота виконана повністю, в заданому обсязі, але деякі свої рішення курсант не зміг чітко обґрунтувати.

65-74 бали – робота виконана в заданому обсязі, але існують прогалини в описі чи обґрунтуванні рішень.

60-64 бали – робота виконана, значна частина рішень не була самостійно обґрунтована курсантом, або потребує значних уточнень.

Рейтингова оцінка трансформується до університетської системи оцінювання згідно з таблицею 1.

Таблиця 1. Переведення рейтингових балів до оцінок за університетською шкалою
Рейтингові бали, RDOцінка за університетською шкалою

Кількість балів	Оцінка
95-100	Відмінно
85-94	Дуже добре
75-84	Добре
65-74	Задовільно
60-64	Достатньо
Менше ніж 60	Незадовільно

9. Додаткова інформація з навчальної дисципліни

Перелік питань, які виносяться на семестровий контроль

Залік проводиться в формі практичного використання курсантами засобів відповідних бібліотек Python до вирішення задач, що стосуються аналізу даних, що описують об'єкти, системи та явища реального світу та самостійного формулювання висновків та прийняття рішень на основі розрахованих та проаналізованих даних. Результати представляються в вигляді аналізу результатів, отриманих в ході виконання обчислень.

Приклади задач, що виносяться на залік.

1. Метод сортування вставкою.
2. Метод сортування вибором.
3. Сортування обміном – бульбашкова та її покращені варіанти.
4. Оцінка обчислювальної складності алгоритму методу сортування вставкою.
5. Метод сортування злиттям.
6. Оцінка обчислювальної складності алгоритму методу сортування злиттям.
7. Структура даних типу піраміди та її формування.
8. Вставка елемента в піраміду.
9. Видалення елемента з піраміди.
10. Сортування на основі пірамід.
11. Швидке сортування Хоара. Розбиття елементів на дві підмножини. Вибір опорних значень.
12. Оцінка обчислювальної складності алгоритму швидкого сортування.
13. Сортування підрахунком та його обчислювальна складність.
14. Порозрядне сортування та його обчислювальна складність.
15. Сортування при рівномірному законі розміщення значень елементів та його обчислювальна складність.
16. Алгоритми пошуку мінімуму серед елементів масиву та його обчислювальна складність.
17. Алгоритми пошуку максимуму серед елементів масиву та його обчислювальна складність.
18. Абстрактні динамічні типи даних, їх елементи. Операції в динамічних типах даних.
19. Стеки та черги.
20. Списки та операції над ними.
21. Поняття бінарного дерева пошуку.
22. Пошук елемента із заданим ключем в бінарному дереві пошуку та його обчислювальна складність.
23. Пошук елемента з мінімальним ключем в бінарному дереві пошуку та його обчислювальна складність.
24. Пошук елемента з максимальним ключем в бінарному дереві пошуку та його обчислювальна складність.
25. Пошук елемента з попереднім значенням ключа в бінарному дереві пошуку та його обчислювальна складність.
26. Пошук елемента з наступним значенням ключа в бінарному дереві пошуку та його обчислювальна складність.
27. Вставка елемента в бінарне дерево пошуку та його обчислювальна складність.
28. Видалення елемента з бінарного дерева пошуку та його обчислювальна складність.
29. Графи та способи їх подання на комп'ютері.
30. Алгоритми формування способів подання графів на комп'ютері за відомим списком суміжності.

31. Алгоритми формування способів подання графів на комп'ютері за відомим списком інцидентності.
32. Алгоритми формування способів подання графів на комп'ютері за відомою матрицею суміжності.
33. Алгоритми формування способів подання графів на комп'ютері за відомою матрицею інцидентності.
34. Алгоритм пошуку в ширину.
35. Алгоритм пошуку в глибину.
36. Обчислювальна складність алгоритму пошуку в ширину.
37. Обчислювальна складність алгоритму пошуку в глибину.
48. Метод пошуку найкоротших шляхів від заданої вершини. Складність алгоритму.