

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Відокремлений структурний підрозділ**  
**«Педагогічний фаховий коледж**  
**Львівського національного університету імені Івана Франка»**

ЗАТВЕРДЖУЮ  
В.о. директора \_\_\_\_\_ О. І. Сурмач  
«\_\_» \_\_\_\_\_ 20\_\_ р.

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**  
**ТЕОРІЯ АЛГОРИТМІВ**

Галузь знань **12 Інформаційні технології**

Спеціальність **122 Комп'ютерні науки**

Статус дисципліни \_\_\_\_\_ нормативна  
Нормативна (вибіркова)

Циклова комісія професійно-орієнтовних дисциплін спеціальності Комп'ютерні науки та математично-природничої підготовки

Дані про вивчення дисципліни

Форма навчання	Курс	Семестр	Загальний обсяг дисциплін	Кількість годин							Вид семестрового контролю	
				Аудиторні заняття					Самостійна робота	Курсова робота	Залік	Екзамен
				Кредити ЄКТС	Разом	Лекції	Лабораторні роботи	Практичні заняття				
Денна	2	4	90/3	48	24	24	-	-	40	-	2	-

Робоча програма складена на основі освітньої програми та навчального плану підготовки **фахового молодшого бакалавра**  
Освітньо-професійний ступінь

Розробник:

\_\_\_\_\_ Лисецька Олександра Юріївна \_\_\_\_\_  
Підпис Ініціали та прізвище викладача (науковий ступінь та вчене звання)

Затверджено на засідання циклової комісії.

Протокол № \_\_\_\_\_ від \_\_\_\_\_ 20\_\_ р.

Голова циклової комісії \_\_\_\_\_ .  
Підпис Ініціали та прізвище

Схвалено на засіданні Педагогічної ради Коледжу.

Протокол № \_\_\_\_\_ від \_\_\_\_\_ 20\_\_ р.

## МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**Мета:** формування у студентів здатностей: аналізувати проблеми алгоритмізації при створенні комп'ютеризованих систем управління та автоматики; виконувати розроблення, аналіз, обґрунтування, оцінювання та реалізацію алгоритмів.

**Предмет:** алгоритм, його складність та трудомісткість.

**Завдання:**

- розширення знань про ролі та місця алгоритмів та структур даних в задачах проектування і реалізації комп'ютеризованих систем управління та автоматики;
- оволодіння вміннями розробляти та оцінювати алгоритми, використовувати структури даних для розроблення алгоритмів;
- засвоєння відомостей про методики оцінювання складності алгоритмів, а також про основні методи уточнення алгоритмів і схеми встановлення невіршуваності масових проблем;
- набуття навичок будувати машини Тюрінга для елементарних функцій та рекурсивні схеми для елементарних функцій;
- дослідження методики встановлення NP-повноти масових проблем та основ автоматизації виробництва алгоритмів.

У результаті засвоєння навчального матеріалу студент повинен

**знати:** Теоретичні, методичні і алгоритмічні основи сучасних інформаційних технологій, загальні принципи побудови ефективних алгоритмів, сучасні методи дослідження та аналізу алгоритмів (понятійно-аналітичний), способи та механізми реалізації ефективних алгоритмів у конкретних застосуваннях.

студенти повинні **вміти:**

1. Реалізовувати основні алгоритми засобами алгоритмічної мови.
2. Розробляти нові математичні методи, ефективні алгоритми і методи реалізації функцій інформаційних систем і технологій у прикладних областях.
3. Аналізувати, теоретично та експериментально досліджувати методи, алгоритми, програми апаратно-програмних комплексів і систем.

4. Створювати та досліджувати математичні та програмні моделі обчислювальних та інформаційних процесів, пов'язаних з функціонуванням об'єктів професійної діяльності.
5. Аналізувати та вибирати обчислювальні методи розв'язання задач проектування інформаційних систем за критеріями мінімізації обчислювальних витрат, стійкості, складності тощо.
6. Проектувати елементи математичного та лінгвістичного забезпечення обчислювальних систем.

### **Програмні компетентності:**

#### **1. Інтегральна компетентність:**

- здатність вирішувати типові спеціалізовані задачі та практичні проблеми у сфері прикладної математики або у процесі навчання, що вимагає застосування положень і методів дискретної математики та математичного аналізу;

- нести відповідальність за результати своєї діяльності;
- здійснювати контроль інших осіб у визначених ситуаціях.

#### **2. Загальні компетентності:**

- здатність до абстрактного мислення, аналізу та синтезу;
- здатність застосовувати знання у практичних ситуаціях;
- здатність генерувати нові ідеї (креативність);
- здатність працювати в команді;
- здатність оцінювати та забезпечувати якість виконуваних робіт.

#### **3. Спеціальні компетентності**

- 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;
- здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного

модельовання предметних областей і створення програмних та інформаційних систем

#### **4. Результати навчання:**

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. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати їх ефективність та складність.

## **Зміст дисципліни:**

### **Змістовний модуль 1. Загальні відомості про алгоритми.**

#### **Вступ до аналізу алгоритмів**

##### **Тема 1. Вступ в теорію алгоритмів.**

Основні поняття теорії алгоритмів. Вимоги до алгоритмів. Історичний огляд, цілі та завдання теорії алгоритмів. Практичне застосування теорії алгоритмів. Формалізація поняття алгоритму. Словесна та графічна форма зображення алгоритму. Блок-схеми. Лінійні алгоритми. Алгоритми розгалуження. Алгоритми з циклом.

##### **Тема 2. Машина Поста.**

Основні поняття та операції. Фінітний 1-процес. Спосіб задання проблеми та формулювання 1. Принцип роботи машини Поста. Двох та трьохстрічкова машини Поста.

##### **Тема 3. Машина Тюрінга та проблеми, які не розв'язуються алгоритмічно.**

Машина Тюрінга. Принцип роботи машини Тюрінга як алгоритму. Проблеми, які не розв'язуються алгоритмічно. Машина Маркова.

##### **Тема 4. Вступ до аналізу алгоритмів.**

Порівняльні оцінки алгоритмів. Псевдокод. Система позначень в аналізі алгоритмів. Класифікація алгоритмів по виду функції трудомісткості. Асимптотичний аналіз функцій.

##### **Тема 5. Трудомісткість алгоритмів та їх часові оцінки**

Елементарні операції в мові запису алгоритмів. Приклади аналізу простих алгоритмів. Перехід до часових оцінок. Приклад поопераційного часового аналізу.

##### **Тема 6. Теорії складності обчислень і класи складності задач**

Теоретична межа трудомісткості завдання. Проблема  $P=NP$ . Клас NPC. Приклади NP-повних задач.

##### **Тема 7. Приклад повного аналізу алгоритму розв'язку задачі про суму**

Формулювання задачі про суму і її асимптотична оцінка. Алгоритм точного розв'язання задачі про суму. Аналіз алгоритму точного розв'язання задачі про суму.

### ***Модульна контрольна робота 1***

#### **Змістовний модуль 2. Алгоритми пошуку. Рекурсивні алгоритми**

##### **Тема 8. Алгоритми пошуку**

Алгоритми пошуку у невідсортованому масиві. Алгоритми пошуку в одновимірному масиві. Робота з векторами. Алгоритми пошуку та сортування у двовимірному масиві. Робота з матрицями.

##### **Тема 9. Рекурсивні функції і алгоритми**

Рекурсивні функції. Рекурсивні процедури. Аналіз трудомісткості рекурсивних алгоритмів методом підрахунку вершин дерева рекурсії. Рекурсивна реалізація алгоритмів. Аналіз трудомісткості алгоритму обчислення факторіала. Розробка рекурсивних алгоритмів.

##### **Тема 10. Рекурсивні алгоритми та методи їх аналізу**

Логарифмічні тотожності. Методи розв'язку рекурсивних співвідношень. Рекурсивні алгоритми. Основна теорема про рекурентні співвідношення.

##### **Тема 11. Прямий аналіз рекурсивного дерева викликів**

Алгоритм сортування злиттям. Злиття відсортованих частин. Merge. Підрахунок вершин в дереві рекурсивних викликів. Аналіз трудомісткості алгоритму сортування злиттям.

##### **Тема 12. Теорія та алгоритми модулярної арифметики**

Алгоритм піднесення числа до цілого степеня. Основні відомості з теорії простих чисел. Функція Ейлера. Алгоритм Евкліда і його аналіз. Аналіз алгоритмів пошуку

##### **Тема 13. Криптосистема RSA і теорія алгоритмів**

Основні поняття з теорії груп та теорії напівгруп. Мультиплікативна група лишків за модулем  $n$ . Степені елементів в  $\mathbf{Z}_n^*$  і пошук великих простих

чисел. Мала теорема Ферма. Теорема Ейлера-Ферма. Криптосистема RSA.  
Криптостійкість RSA і складність алгоритмів факторизації.

*Модульна контрольна робота 2*

*Залікове заняття*

## 2. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назва теми	Кількість годин			
	Денна форма навч.			
	Усього	Лекції	Лабораторні роботи	СРС
<b>IV семестр</b>				
<b>Змістовий модуль I. Загальні відомості про алгоритми. Вступ до аналізу алгоритмів</b>				
Тема 1. Вступ в теорію алгоритмів	6	2	2	2
Тема 2. Машина Поста	6	2	2	2
Тема 3. Машина Тюрінга	8	2	2	4
Тема 4. Вступ до аналізу алгоритмів	6	2	2	2
Тема 5. Трудомісткість алгоритмів та їх часові оцінки	6	2		4
Тема 6. Теорії складності обчислень і класи складності задач	5	2		3
Тема 7. Приклад повного аналізу алгоритму розв'язку задачі про суму	4	2		2
<i>Модульна контрольна робота 1</i>	2		2	
<b>Змістовий модуль II. Алгоритми пошуку. Рекурсивні алгоритми</b>				
Тема 8. Алгоритми пошуку	12		6	6
Тема 9. Рекурсивні функції і алгоритми	8	2	2	4
Тема 10. Рекурсивні алгоритми та методи їх аналізу	2	2		
Тема 11. Прямий аналіз рекурсивного дерева викликів	6	2	2	2
Тема 12. Теорія та алгоритми модулярної арифметики	7	2	2	3
Тема 13. Криптосистема RSA і теорія алгоритмів	8	2		6
<i>Модульна контрольна робота 2</i>	2		2	
<i>Залік</i>	2		2	



Разом за IV семестр	90	24	26	40
---------------------	----	----	----	----

**3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**  
**Зміст лекційного курсу для студентів денної форми навчання**  
**IV семестр**

№ з/п	Перелік тем лекцій	Кількість годин
1.	<p><b>Тема . Вступ в теорію алгоритмів.</b></p> <p>Основні поняття теорії алгоритмів. Вимоги до алгоритмів. Історичний огляд, цілі та завдання теорії алгоритмів. Практичне застосування теорії алгоритмів. Формалізація поняття алгоритму. Словесна та графічна форма зображення алгоритму. Блок-схеми.</p>	2
2.	<p><b>Тема. Машина Поста.</b></p> <p>Основні поняття та операції. Фінітний 1-процес. Спосіб задання проблеми та формулювання 1. Принцип роботи машини Поста. Двох та трьохстрічкова машини Поста.</p>	2
3.	<p><b>Тема. Машина Тюрінга та проблеми, які не розв'язуються алгоритмічно.</b></p> <p>Машина Тюрінга. Принцип роботи машини Тюрінга як алгоритму. Проблеми, які не розв'язуються алгоритмічно. Машина Маркова.</p>	2
4.	<p><b>Тема. Вступ до аналізу алгоритмів.</b></p> <p>Порівняльні оцінки алгоритмів. Псевдокод. Система позначень в аналізі алгоритмів. Класифікація алгоритмів по виду функції трудомісткості. Асимптотичний аналіз функцій. Лінійні алгоритми. Алгоритми розгалуження. Алгоритми з циклом.</p>	2

5.	<p><b>Тема. Трудомісткість алгоритмів та їх часові оцінки</b></p> <p>Елементарні операції в мові запису алгоритмів. Приклади аналізу простих алгоритмів. Перехід до часових оцінок. Приклад поопераційного часового аналізу.</p>	2
	<p><b>Тема. Теорії складності обчислень і класи складності задач</b></p> <p>Теоретична межа трудомісткості завдання. Проблема <math>P=NP</math>. Клас NPC. Приклади NP-повних задач.</p>	2
	<p><b>Тема. Приклад повного аналізу алгоритму розв'язку задачі про суму</b></p> <p>Формулювання задачі про суму і її асимптотична оцінка. Алгоритм точного розв'язання задачі про суму. Аналіз алгоритму точного розв'язання задачі про суму.</p>	2
	<p><b>Тема. Рекурсивні функції і алгоритми</b></p> <p>Рекурсивні функції. Рекурсивні процедури. Аналіз трудомісткості рекурсивних алгоритмів методом підрахунку вершин дерева рекурсії. Рекурсивна реалізація алгоритмів. Аналіз трудомісткості алгоритму обчислення факторіала. Розробка рекурсивних алгоритмів.</p>	2
	<p><b>Тема. Рекурсивні алгоритми та методи їх аналізу</b></p> <p>Логарифмічні тотожності. Методи розв'язку рекурсивних співвідношень. Рекурсивні алгоритми. Основна теорема про рекурентні співвідношення.</p>	2
	<p><b>Тема. Прямий аналіз рекурсивного дерева викликів</b></p> <p>Алгоритм сортування злиттям. Злиття відсортованих частин. Merge. Підрахунок вершин в дереві рекурсивних викликів. Аналіз трудомісткості алгоритму сортування злиттям.</p>	2

	<b>Тема . Теорія та алгоритми модулярної арифметики</b> Алгоритм піднесення числа до цілого степеня. Основні відомості з теорії простих чисел. Функція Ейлера. Алгоритм Евкліда і його аналіз. Аналіз алгоритмів пошуку	2
	<b>Тема. Криптосистема RSA і теорія алгоритмів</b> Основні поняття з теорії груп та теорії напівгруп. Мультиплікативна група лишків за модулем $n$ . Степені елементів в $Z_n^*$ і пошук великих простих чисел. Мала теорема Ферма. Теорема Ейлера-Ферма. Криптосистема RSA. Криптостійкість RSA і складність алгоритмів факторизації.	2
	Разом за IV семестр	24

**Перелік практичних (лабораторних, семінарських) занять  
для студентів денної форми навчання  
IV семестр**

№ з/п	Тема практичного (лабораторного, семінарського) заняття	Кількість годин
1.	Словесна форма представлення алгоритму. Графічна форма зображення алгоритму. Зображення алгоритму на псевдокодi	2
2.	Складання програм для машини Поста	2
3.	Складання програм для машини Тюрінга	2
4.	Розв'язання задач на складання лінійних, циклічних алгоритмів, а також алгоритмів розгалуження	2
	<i>Модульна контрольна робота 1.</i>	2
5.	Алгоритми пошуку у неупорядкованому масиві. Деякі методи розв'язання типових задач в одновимірному та двовимірному масивах	6
6.	Розробка рекурсивних алгоритмів	2
7.	Розв'язання задач на визначення складності алгоритму	2
8.	Аналіз алгоритмів пошуку	2
	<i>Модульна контрольна робота 2.</i>	2

	<i>Залік</i>	2
	<b>Разом за IV семестр</b>	<b>26</b>

### Самостійна робота

Номер тижня	Тема самостійної роботи	Кількість годин
1.	Зображення алгоритму за допомогою UML-діаграм.	2
2.	Багатострічкові машини Поста.	2
3.	Машина Маркова.	4
4.	Порівняння трудомісткості лінійного та циклічного алгоритму піднесення до квадрату	2
5.	Поопераційний аналіз алгоритму обчислення факторіалу.	4
6.	Приклади задач NPC, а також NP задач, які не є повними.	3
7.	Різновиди алгоритмів розв'язання задачі про суму. Задачі рюкзака та комівояжера.	2
8.	Зображення векторів і матриць у програмуванні. Порівняння алгоритмів сортування одновимірного масиву. Робота з матрицею, як з двовимірним масивом. Трудомісткість алгоритму транспонування матриці.	6
9.	Рекурсивний алгоритм обчислення факторіала. Рекурентні рівняння.	4
10.	Основні типи графів. Дерева. Ліс. Зображення графів у програмування.	2
11.	Решето Ератосфена. Аналіз алгоритму.	3
12.	Застосування RSA алгоритму. Електронний цифровий підпис.	6
<b>Разом за IV семестр</b>		<b>40</b>

#### 4. ПИТАННЯ, ЯКІ ВИНЕСЕНО НА ЗАЛІК З ДИСЦИПЛІНИ

##### Теоретичні питання:

1. Основні поняття теорії алгоритмів.
2. Вимоги до алгоритмів.
3. Історичний огляд, цілі та завдання теорії алгоритмів.
4. Практичне застосування теорії алгоритмів.

5. Формалізація поняття алгоритму.
6. Словесна та графічна форма зображення алгоритму. Блок-схеми.
7. Лінійні алгоритми.
8. Алгоритми розгалуження.
9. Алгоритми з циклом.
10. Машина Поста
11. Машина Тюрінга
12. Порівняльні оцінки алгоритмів.
13. Псевдокод. Система позначень в аналізі алгоритмів.
14. Класифікація алгоритмів по виду функції трудомісткості.
15. Елементарні операції в мові запису алгоритмів.
16. Перехід до часових оцінок.
17. Теоретична межа трудомісткості завдання.
18. Проблема  $P=NP$ . Клас NPC. Приклади NP-повних задач.
19. Формулювання задачі про суму.
20. Асимптотична оцінка задачі про суму.
21. Аналіз алгоритму точного розв'язання задачі про суму.
22. Алгоритми пошуку.
23. Рекурсивні функції.
24. Рекурсивні процедури.
25. Аналіз трудомісткості рекурсивних алгоритмів методом підрахунку вершин дерева рекурсії.
26. Рекурсивна реалізація алгоритмів.
27. Аналіз трудомісткості алгоритму обчислення факторіала.
28. Методи розв'язку рекурсивних співвідношень.
29. Рекурсивні алгоритми.
30. Основна теорема про рекурентні співвідношення.
31. Алгоритм сортування злиттям.
32. Злиття відсортованих частин. Merge.
33. Аналіз трудомісткості алгоритму сортування злиттям.
34. Алгоритм піднесення числа до цілого степеня.

35. Алгоритм Евкліда і його аналіз.
36. Мала теорема Ферма.
37. Теорема Ейлера-Ферма.
38. Криптосистема RSA.
39. Криптостійкість RSA.
40. Складність алгоритмів факторизації.

## Практичні завдання

**Завдання 1.** Визначити площу трапеції по заданих введених значеннях її основ ( $a$  та  $b$ ) і висоти ( $h$ ).

**Завдання 2.** Визначити середнє арифметичне двох чисел, якщо  $a$  додатне і частку  $\left(\frac{a}{b}\right)$ , в протилежному випадку.

**Завдання 3.** Скласти алгоритм знаходження суми цілих чисел в діапазоні від 1 до 10.

**Завдання 4.** Лінійний алгоритм, виконаний із попередніх завдань, переробити на циклічний із заданою кількістю циклів.

**Завдання 5.** Знайти суму значень змінної  $P$ , вважаючи, що її початкове значення дорівнює нулю, тобто  $P = 0$ . В кожному циклі змінна збільшується на 2, тобто  $P := P + 2$ . Кількість циклів рівна шести. В результаті даного алгоритму значення змінної має дорівнювати  $P = 12$ .

**Завдання 6.** Нехай задано початкові значення змінних:

$$x := 1;$$

$$y := 5.$$

А також задано такий цикл:

*while*  $y > x$  *do*:

$$y := y - x$$

Визначити кількість циклів і значення змінних  $x$ ,  $y$  після виходу з циклу.

**Завдання 7.** Скласти алгоритм розв'язку задачі за допомогою алгоритмічної мови псевдокоду і за допомогою блок-схем, використовуючи конструкцію лінійного алгоритму.

1. Обчислити площу поверхні та об'єм зрізаного конуса за формулами:

$$S = \pi (R + r)l + \pi R^2 + \pi r^2 ;$$

$$V = \frac{1}{3} \pi (R^2 + r^2 + Rr)h .$$

2. Обчислити координати центру маси трьох матеріальних точок з масами  $m_1, m_2, m_3$  і координатами  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  за формулами:

$$x_c = \frac{(m_1 x_1 + m_2 x_2 + m_3 x_3)}{(m_1 + m_2 + m_3)} ;$$

$$y_c = \frac{(m_1 y_1 + m_2 y_2 + m_3 y_3)}{(m_1 + m_2 + m_3)} .$$

3. Обчислити площу трикутника зі сторонами  $a, b, c$  за формулою Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

де  $p$  – півпериметр, який обчислюється за формулою

$$\frac{a + b + c}{2} .$$

**Завдання 8.** Нехай  $A = \{0,1\}$ . На першій стрічці двострічкової машини Поста задане слово. Скопіювати це слово у зворотному порядку на другу стрічку.

**Завдання 9.** Нехай  $A = \{0,1\}$ . Два двійкові числа записані на першій та другій стрічках трьохстрічкової машини Поста, відповідно. Вважаючи, що обидва слова записані нормально (вони завжди існують і, за винятком числа 0, починаються з одиниці). Визначити суму цих двох чисел і записати результат на третю стрічку.

**Завдання 10.**  $A = \{a, b\}$ . У вхідному слові замінити всі символи  $a$  на  $b$ .

**Завдання 11.**  $A = \{01\}$ . Вкінці вхідного слова записати 1. Якщо слово порожнє, то вивести повідомлення про помилку (відмітивши цей стан  $\Omega$ ).

**Завдання 12.**  $A = \{0,1,2\}$ . У вхідному слові замінити всі комбінації «012» на зірочки «\*».

**Завдання 13.** Обчислити об'єм  $V$  і площу бічної поверхні циліндра  $S$  за введеними значеннями радіуса основи  $R$  і висоти циліндра  $H$ .

$$V = \pi R^2 H ;$$

$$S = 2\pi R H .$$



**Завдання 14.** Обчислити об'єм  $V$  і площу бічної поверхні конуса  $S$  за введеними значеннями радіуса основи  $r$ , висоти  $h$  і твірної  $l$ :

$$V = \frac{1}{3}\pi r^2 h;$$

$$S = \pi r l.$$

**Завдання 15.** Скласти програму для розв'язку квадратного рівняння  $ax^2 + bx + c = 0$ .

**Завдання 16.** Визначити максимальне парне число із двох введених.

**Завдання 17.** Визначити, чи можна із відрізків довжиною  $x$ ,  $y$  та  $z$  побудувати трикутник.

**Завдання 18.** Ввести два числа  $a$  та  $b$ . Більше число замінити потрійним добутком, менше – півсумою.

**Завдання 19.** Якщо серед трьох чисел  $a$ ,  $b$ ,  $c$  є хоча б одне парне, то знайти максимальне число, інакше – мінімальне. **Завдання 20.** Визначити, у якому квадранті знаходиться точка з координатами  $(x, y)$  і вивести номер квадранта на екран.

**Завдання 21.** Знайти суму чисел, кратних 3, в діапазоні від 0 до 50.

**Завдання 22.** Знайти суму перших десяти чисел, кратних п'яти.

**Завдання 23.** Знайти добуток парних чисел в діапазоні від 2 до 30.

**Завдання 24.** Вводяться додатні числа. Зупинити ввід, коли сума введених чисел буде більшою, ніж 100.

**Завдання 25.** Необхідно знайти суму чисел, кратних 7, в діапазоні від 0 до 100. Вивести на екран суму чисел і їх кількість.

**Завдання 26.** Визначити кількість цілих чисел, кратних 3 (від 3 і далі), що дають в сумі число, яке є більшим за 200.

**Завдання 27.** Визначити порядкові номери нулів у масиві.

**Завдання 28.** Всі від'ємні елементи масиву замінити нулями. Видати повідомлення про кількість таких замін.

**Завдання 29.** Знайти порядковий номер мінімального числа в масиві.

**Завдання 30.** Знайти значення і координати першого по порядку додатного елемента в масиві.

**Завдання 31.** Знайти кількість одиниць в масиві.

**Завдання 32.** Створіть програми, які реалізують рекурсивний та інтервальний алгоритми обчислення  $n$ -го числа Фібоначчі. Порівняйте час їх роботи для  $n=10, 20, 30, \dots, 90$ . Щоб уникнути переповнення, для  $n < 100$  використовуйте беззнакове зображення чисел в 8 байт. Результати порівнянь оформити у вигляді таблиці та графіка.

**Завдання 33.** Проаналізуйте обчислювальну та просторову складність алгоритмів множення чисел. На основі аналізу проведіть порівняння алгоритмів.

**Завдання 34.** Розробіть два різні за складністю алгоритми піднесення до цілого невід'ємного степеня  $a^n, n \in \mathbb{Z}^+$ . Зверніть увагу, що для обчислення  $a^{15}$  може знадобитися 6 операцій множення

$$a^{15} = a(a^7)^2 = a(a(a^3)^2)^2 = a(a(a(a^2))^2)^2$$

А для обчислення  $a^{100}$  може знадобитися всього 14 операцій множення. Для розроблених алгоритмів визначте обчислювальну складність та проведіть порівняльний аналіз.

**Завдання 35.** За допомогою виразу

$$\begin{pmatrix} F_n \\ F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_{n-1} \\ F_{n-2} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{n-1} \begin{pmatrix} F_1 \\ F_0 \end{pmatrix}$$

розробіть алгоритм для обчислення  $n$ -го числа Фібоначчі, який працює за  $O(\ln n)$  кроків.

**Завдання 36.**

- 1) Згенеруйте масив цілих випадкових чисел розміром  $N = 1000$ . (Для генерації випадкових чисел скористайтесь класом *System.Random*).
- 2) Відсортуйте отриманий масив довільним методом сортування.
- 3) Введіть з клавіатури деяке ціле число.
- 4) Користуючись методом інтерполяційного пошуку, визначте, позицію введеного числа в масиві, якщо воно в ньому присутнє.

**Завдання 37.** Складіть програму, яка в заданому текстовому файлі шукає всі входження, введеного користувачем слова, формуючи список позицій (від початку файлу) шуканого слова. Метод пошуку – довільний, крім лінійного пошуку підрядка.

## 5. МЕТОДИ НАВЧАННЯ

- застосування інтерактивних методів навчання;
- проведення лекційних занять та лабораторних робіт;
- впровадження комп'ютерної технології навчання;
- зустрічі з науковцями, політиками, діячами культури.

## 6. МЕТОДИ КОНТРОЛЮ

- поточний контроль знань студентів;
- поточний контроль самостійної роботи студентів;
- усний захист лабораторних робіт;
- самостійна підготовка до лабораторних робіт;
- участь у дискусіях;
- тестовий контроль;
- оцінювання та аналіз модульних контрольних робіт;
- залік.

## 7. ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Оцінювання знань студентів здійснюється за 100-бальною шкалою. Максимальна кількість балів при оцінюванні знань за поточну успішність становить 1000 балів.

Результат роботи студента протягом семестру (100 балів) складається з наступних компонентів:

- результатів виконання та захисту *лабораторних робіт* (30 балів). Підсумкова кількість балів за роботу на лабораторних роботах (24 години) визначається як сума балів, отриманих за усні чи письмові відповіді студента;
- *двох модульних контрольних робіт* (20 балів + 20 балів = 40 балів), що проводиться за результатами вивчення частини навчального курсу (змістового модуля 1 та змістового модуля);

- **заліку** (30 балів), що підсумовує знання основних теоретичних положень курсу, студент складає під час залікової сесії **в письмовій формі на платформі Moodle або Zoom.**

Під час складання іспиту на платформі **Moodle або Zoom** студент протягом 90 хвилин повинен дати відповідь на 3 теоретичні запитання та розв'язати 3 задачі. Кожне правильно виконане завдання оцінюється в 5 балів. Максимальна кількість балів за залік – 30.

Поточне тестування та самостійна робота					Сума	
Змістовий модуль 1			Змістовий модуль 2			
Т 1 – Т 3	Т 4 – Т 7	<b>МКР 1</b>	Т 8 – Т 13	<b>МКР 2</b>	Підсумковий тест (залік)	
9	9	20	12	20	<b>30</b>	<b>100</b>

#### Шкала оцінювання: національна та ЄКТС

Оцінка в балах	ECTS	Визначення	Екзаменаційна оцінка, оцінка з диференційованого заліку (національна шкала)
<b>90-100</b>	<b>A</b>	Відмінно	Відмінно
<b>81-89</b>	<b>B</b>	Дуже добре	Добре
<b>71-80</b>	<b>C</b>	Добре	
<b>61-70</b>	<b>D</b>	Задовільно	Задовільно
<b>51-60</b>	<b>E</b>	Достатньо	

#### 8. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

1. Робоча навчальна програма дисципліни.
2. Навчальні-наочні посібники, технічні засоби навчання.
3. Конспекти лекцій з основ теорії алгоритмів.
4. Комплекс завдань для модульних контрольних робіт.
5. Інструктивно-методичні матеріали до лабораторних робіт.
6. Перелік питань до заліку.
7. Методичні вказівки, рекомендації та розробки викладача.

8. Методичні матеріали, що забезпечують самостійну роботу студентів.

## **9. РЕКОМЕНДОВАНА ЛІТЕРАТУРА**

### **Основна:**

1. Кнут Д. Искусство программирования, том 1. Основные алгоритмы — 3-е изд. — М.: «Вильямс», 2006. — 720 С. . — ISBN 0-201-89683-4
2. Ильиных А.П. Теория алгоритмов. Учебное пособие. – Екатеринбург, 2006. - 149 с.
3. Марков А.А., Нагорный Н.М. Теория алгорифмов. – М.: Наука, 1984. -432 с.

### **Допоміжна:**

1. Мальцев А.И. Алгоритмы и рекурсивные функции. – М.: Наука, 1986.
2. Кормен Томас Х., Лейзерсон Чарльз И., Ривест Рональд Л., Штайн К. Алгоритмы: построение и анализ — 2-е изд. — М.: «Вильямс», 2006. — 1296 С. — ISBN 0-07-013151-1

## **11. ІНФОРМАЦІЙНІ РЕСУРСИ**

1. Рыжова Н.И., Голанова А.В., Швецкий М.В., Луценко А.Ю. Теория алгоритмов (электронный учебник) // <http://ric.uni-altai.ru/Fundamental/teor-alg/>