



Алгоритмізація та програмування.

Курсова робота

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Освітня програма	<i>Інтелектуальні сервіс-орієнтовані розподілені обчислювання</i>
Статус дисципліни (код)	<i>Нормативна</i>
Форма навчання	<i>очна(денна)/дистанційна/змішана</i>
Рік підготовки, семестр	<i>1 курс, весняний семестр</i>
Обсяг дисципліни	<i>1 кредит ЄКТС. Самостійна робота студентів – 30 годин</i>
Семестровий контроль/ контрольні заходи	<i>Залік (у формі усного захисту курсової роботи)</i>
Розклад занять	<i>Rozklad.kpi.ua</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Романов Валерій Володимирович, v.romanov2516@gmail.com к.т.н. Безносик Олександр Юрійович, beznosyk.oleksandr@ill.kpi.ua к.е.н., доцент Просянкін-Жарова Тетяна Іванівна, t.pruman@gmail.com</i>
Розміщення курсу	<i>https://do.ipu.kpi.ua/course/view.php?id=393</i>

Програма навчальної дисципліни

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

Дисципліна "Алгоритмізація та програмування. Курсова робота" – це індивідуальна самостійна робота студентів з елементами наукового дослідження. Її метою є закріплення та поглиблення навичок та умінь, набутих в результаті вивчення освітньої компоненти «Алгоритмізація та програмування» та опанування навичок виконання завдань з розробки завершеного програмного продукту мовою програмування високого рівня C/C++ з використанням сучасних технологій розробки програмного забезпечення та оформлення відповідної супровідної документації.

Завданням курсової роботи є закріплення вмінь самостійно спроектувати загальний алгоритм роботи складної системи, провести необхідні обґрунтування для розробки програмних засобів, підібрати оптимальні алгоритми для реалізації окремих складових, скористатись для цього сучасними джерелами інформації, у тому числі Інтернет-ресурсами, самостійно обрати середовище розробки та обґрунтувати цей вибір; самостійно реалізувати побудовані алгоритми програмно, за потреби, використати готові алгоритми та довести роботу над програмним забезпеченням до логічного кінця, проаналізувати отримані результати, ознайомлення з основними етапами розробки завершеного програмного продукту, готового до використання споживачем та основними вимогами до оформлення текстового, програмного та ілюстративного матеріалу звіту до нього. Знання та вміння, здобуті в результаті виконання курсової роботи, в подальшому застосовуватимуться студентом при виконання наступних курсових та дипломних робіт, індивідуальних навчально-дослідних та навчально-виробничих завдань, наукових досліджень тощо.

Програмні результати навчання:

Загальні компетентності

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 і 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.

ЗК 12 Здатність оцінювати та забезпечувати якість виконуваних робіт.

Фахові компетентності спеціальності

ФК 3 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 моделювання предметних областей і створення програмних та інформаційних систем.

ФК 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.

Програмні результати навчання

ПРН 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.

ПРН 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.

ПРН 9 Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

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

Курсова робота з «Алгоритмізації та програмування» вивчається у другому семестрі, тому для успішного засвоєння дисципліни необхідні знання курсу «Алгоритмізація та програмування» математики та інформатики на рівні, вищому за «базовий» (для першого семестру) та дискретна математика, математичний аналіз, аналітична геометрія та лінійна алгебра, алгоритмів і структур даних.

На результатах навчання з дисципліни базуються такі дисципліни, як «Проектування та аналіз обчислювальних алгоритмів», «Об'єктно-орієнтоване програмування».

Базова література

1. Трофименко О.Г. Трофименко, Ю.В. Прокоп, Н.І. Логінова, О.В. Задерейко С++. Алгоритмізація та програмування : підручник . 2-ге вид. перероб. і доповн. Одеса. Фенікс, 2019.
2. Васильєв, О. Програмування на С++ в прикладах і задачах :навчальний посібник. Київ. Видавництво Ліра-К, 2020.
3. Тверитникова О. Є. Крилова В. А., Васильченков О. Г. Базові алгоритми та основи програмування: теорія і практика: навчальний посібник. Харків. НТУ "ХПІ" ; А.М. Панов, 2020.
4. Ковалюк, Т. В. Алгоритмізація та програмування: підручник. Львів. Видавництво "Магнолія 2006", 2019.
5. Козак, Л. І. Костюк І. В., Стасевич С. П. Основи програмування: навчальний посібник. Львів. Новий Світ – 2000, 2019.

Додаткова література

6. Белов Ю. А., Карнаух Т. О., Коваль Ю. В., Ставровський А. Б. Вступ до програмування мовою С++. Організація обчислень: навч. посіб. Київ. Видавничо-поліграфічний центр "Київський університет", 2012. 175 с.

3. Методика опанування навчальної дисципліни(освітнього компонента)

Зміст навчальної дисципліни

Студенти самостійно вибирають тему курсової роботи із переліку запропонованих або можуть запропонувати свою власну тему після обговорення її з керівником, виходячи із особистих схильностей, інтересів та знань.

Тематика (перелік можливих тем) курсових робіт представлена у силабусі дисципліни.

Вибір теми підтверджується оформленням (разом з керівником) технічного завдання до виконання курсової роботи. Не допускається виконання курсових робіт на однакову тему різними студентами. У випадку виникнення потреби змінити тему курсової роботи рішення приймається викладачем на підставі мотивованої заяви. Рішення приймається не пізніше ніж за місяць до планового терміну захисту курсових робіт.

Тематика курсових робіт може бути розділена на окремі групи, кожна із яких має свої особливості при реалізації і орієнтована на застосування спеціальних знань з інших дисциплін, що викладаються під час виконання курсової роботи або, які вже були викладені у першому семестрі.

№ п/п	Змістові модулі / теми	Кількість годин, відведених на:			Термін виконання
		лекції	лаб. заняття	сам. робота.	
1	2	3	4	5	
1	Підготовчий етап. Ознайомлення з тематикою курсових робіт. Вибір теми курсової роботи			4	1-й та 2-й тижні
2	Аналіз предметної області задачі, постановка задачі і вибір методів та інструментів розробки			6	3—й- 5-й тижні
3	Побудова алгоритму розв'язку задачі та вибір відповідних структур даних, проектування інтерфейсу			6	6-й – 9-й тижні
4	Програмна реалізація, .тестування програми і аналіз результатів			6	10-й -13-й тижні
5	Оформлення пояснювальної записки			6	14-й - 17-й тижні
6	Підготовка до захисту курсової роботи, захист курсової роботи			2	18-й тиждень
	Всього модуль			30	

4. Самостійна робота студента

T1	Вибір теми курсової роботи	2
T2	Оформлення завдання	2
T3	Аналіз предметної області з використанням відповідних літературних джерел та інтернет-ресурсів	2
T4	Формулювання постановки задачі	2
T5	Обґрунтування вибору інструментів розробки з використанням відповідних літературних джерел та інтернет-ресурсів	2
T6	Вибір та дослідження методів, використовуваних у розробці вибір відповідних структур даних	2
T7	Проектування алгоритму	2
T8	Проектування інтерфейсу	2
T9	Програмна реалізація основних функцій програми	2
T10	Програмна реалізація інтерфейсу програми, розробка інструкції користувача	2
T11	Тестування програми і аналіз результатів.	2

T12	Оформлення пояснювальної записки: форматування тексту розділів	2
T13	Оформлення пояснювальної записки: оформлення графічного матеріалу	2
T14	Оформлення пояснювальної записки: оформлення додатків та списку використаних джерел	2
T15	Підготовка до захисту курсової роботи, захист курсової роботи	2
	Всього	30

Політика та контроль

5. Політика навчальної дисципліни (освітнього компонента)

Вимоги, яких має дотримуватися студент в рамках даної дисципліни:

- самостійно виконувати всі завдання передбачені змістом навчальної дисципліни;
- дотримуватись календарного плану виконання робіт
- дотримуватись політики академічної доброчесності, визначеної у Кодексі честі Національного технічного університету України «Київський політехнічний інституті мені Ігоря Сікорського».

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

Семестровий контроль: залік у формі усного захисту курсової роботи.

PCO курсової роботи має дві складові:

• *стартова* – характеризує якість пояснювальної записки, текстового та графічного (ілюстративного) матеріалу: дотримання встановленого графіка виконання курсової роботи, сучасність та обґрунтування прийнятих рішень, правильність застосування методів аналізу і розрахунку, якість оформлення, виконання вимог нормативних документів, якість графічного матеріалу і дотримання вимог стандартів тощо. Розмір стартової складової дорівнює **60 балів**;

• *складова захисту* – характеризує якість захисту курсової роботи: якість доповіді, ступінь володіння матеріалом, ступінь обґрунтування прийнятих рішень, вміння захищати свою думку, відповідей на запитання членів комісії з проведення семестрового контролю тощо - **40 балів**;

Система і критерії оцінювання характеристик стартової складової та складової захисту.

- *стартова складова:*
 - *якість аналізу предметної області – 5 балів;*
 - *якість обґрунтування вибору інструментів розробки та формулювання постановки задачі - 5 балів;*
 - *повнота і коректність опису методів розробки – 5 балів;*
 - *якість обґрунтування використання даних та їх представлення в програмі – 5 балів;*
 - *якість проектування алгоритму – 10 балів;*
 - *якість розробки інтерфейсу – 10 балів;*
 - *якість та складність написання коду програми – 10 балів;*
 - *якість тестування розробленої програми – 5 балів;*
 - *якість оформлення пояснювальної записки – 5 балів;*
- *складова захисту курсової роботи:*
 - *ступінь володіння теоретичним матеріалом та термінологією – 15 балів*
 - *ступінь володіння алгоритмом програми та розуміння особливостей роботи програми, вміння внести зміни в код – 15 балів;*
 - *вміння презентувати розробку – 10 балів.*

Після захисту курсової роботи комісія з проведення семестрового контролю підсумовує бали за стартовою складовою та складовою захисту, зводить до рейтингової оцінки та переводить до оцінок за університетською шкалою.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

7. Додаткова інформація з дисципліни (освітнього компонента)

Орієнтовний перелік тем курсових робіт:

№	Тема	Обов'язкове завдання
1.	Динамічні лінійні структури даних (списки, черги, стеки)	Розробка пакету програм для роботи з динамічними структурами даних. Передбачити розвинену систему меню з вибору типу задач. Результати розв'язання задач подавати із графічним зображенням та текстовими коментарями.
2.	Лінійні структур даних (розробка, тестування)	Розробити функції для створення, включення і вибірки елемента: а) черги векторної структури; б) черги з пріоритетами векторної структури; в) черги з пріоритетами одно зв'язної спискової структури; Організувати інтерфейс роботи з функціями.
3.	Лінійні структур даних (розробка, тестування)	Розробити функції для створення, включення і вибірки елемента: а) стека одно зв'язної спискової структури; б) стека двох зв'язної спискової структури; в) черги з пріоритетами двох зв'язної спискової структури; Організувати інтерфейс роботи з функціями.
4.	Лінійні структур даних (розробка, тестування)	Визначити число елементів в лінійному списку; Розробити функцію конкатенації двох лінійних списків; З упорядкованого списку виключити всі елементи, значення ключів яких знаходяться в заданих межах. З виключених елементів утворити новий список. Організувати інтерфейс роботи з функціями.
5.	Динамічні нелінійні структури даних (бінарні дерева, AVL-дерева, B-дерева)	Розробка пакету програм для роботи з динамічними структурами даних. Передбачити розвинену систему меню з вибору типу задач. Результати розв'язання задач подавати із графічним зображенням та текстовими коментарями.

6.	Бінарне дерево та його обробка	Програмна реалізація перегляду бінарних дерев. Перегляд у прямому, симетричному та зворотному порядку. Визначити листя дерева і виділити їх, за заданим елементом відновити відповідну частку дерева, визначити гілку дерева, якій належить заданий елемент .
7.	Бінарне ідеально-збалансоване дерево	Побудова бінарного ідеально-збалансованого дерева и реалізація основних операцій з ним. Організувати інтерфейс роботи
8.	Динамічні нелінійні структури даних (розробка, тестування)	Розробити рекурсивну функцію для визначення висоти бінарного дерева; Розробити функцію для підрахунку листів в бінарному дереві; в) розробити функцію для визначення в бінарному дереві кількості вузлів з одним дочірнім вузлом. Організувати інтерфейс роботи з функціями.
9.	Хешування та хеш-таблиці	Вибір функцій хешування. Вибір функцій хешування для рядків. . Розв'язання колізій шляхом лінійного зондування при роботі з хеш-таблицями. Програмна реалізація.
10.	Хешування та хеш-таблиці	Таблиця містить дані про успішність студентів, в якості ключа використовується прізвище студента. Вибрати функцію розстановки, розробити функції включення, видалення та пошуку запису по ключу: а) хеш-таблиця із зовнішніми ланцюжками, рядок таблиці містить перший запис з даного входу і покажчик початку зовнішньої ланцюжка з даного входу; б) те ж, але рядок таблиці містить тільки покажчик початку зовнішнього ланцюжка; в) хеш-таблиця із зовнішньою таблицею переповнення;
11.	Хешування та хеш-таблиці	Таблиця містить дані про успішність студентів, в якості ключа використовується прізвище студента. Вибрати функцію розстановки, розробити функції включення, видалення та пошуку запису по ключу а) хеш-таблиця з внутрішніми ланцюжками; записи, що викликали колізії, вставляються в вільні комірки з початку таблиці; б) таблиця з лінійною відкритою адресацією, $z = 1$; в) хеш-таблиця з квадратичною відкритою адресацією; г) хеш-таблиця з подвійним хешуванням.

12.	Бінарні дерева виразів	Реалізація алгоритмів обчислення значень виразів в різних формах записи. Префіксна, постфіксна, інфіксна форми записів. Передбачити розвинену систему меню з вибору типу форм.
13.	Застосування бінарних дерев	Сортування, пошук, кодування інформації – реалізація алгоритму Хафмана. Передбачити розвинену систему меню з вибору типу задач.
14.	Розробка пакету програм пошуку підрядка в рядку.	Розробка пакету програм. Реалізація алгоритмів Кнута-Моріса-Прата, Бойера-Мура, Карпа-Рабіна. Передбачити розвинену систему меню з вибору типу алгоритмів Провести порівняльний аналіз алгоритмів..
15.	Хешування даних	Програмна реалізація методів розв'язування колізій при хешуванні.
16.	Рекурсивні визначення та функції	Розробка пакету програм рекурсивних визначень та функцій. Малювання само подібних ломаних. Передбачити розвинену систему меню з вибору функцій.
17.	Довга ціле численна арифметика	Розробка пакету програм по нестандартній обробки чисел. Адитивні операції.
18.	Довга ціле численна арифметика	Розробка пакету програм по нестандартній обробки чисел. Мультиплікаційні операції.
19.	Сортування даних	Розробка пакету програм по реалізації алгоритмів сортування. Розгляд алгоритмів порозрядного, блочного сортування, сортування підрахунком. Провести аналіз ефективності алгоритмів. Довести, що складність алгоритмів $O(n)$, $O(n+k)$.
20.	Швидке сортування даних	Розробка пакету програм по реалізації швидких алгоритмів сортування. Розгляд алгоритмів QuickSort, MergeSort, Heapsort (пірамідальне сортування). Провести аналіз ефективності алгоритмів. Довести, що складність алгоритмів $O(n \log n)$.
21.	Швидке сортування даних	Розробка пакету програм з реалізації швидких алгоритмів сортування. Розгляд алгоритмів CombSort, Smoothsort, Introsort (сполучення пірамідального та швидкого сортування). Провести аналіз ефективності алгоритмів. Довести, що складність алгоритмів $O(n \log n)$.
22.	Обхід графів	Розробка пакету програм по реалізації алгоритмів обходу графів
23.	Обхід графів	Розробка пакету програм з використання алгоритмів обходу графів
24.	Робота зі зважуваними графами	Розробка пакету програм з реалізації алгоритмів Дейкстри, Флойда та їх використання

25.	Обчислювальна геометрія на площині	Розробка пакету програм з реалізації алгоритмів обчислювальної геометрії на площині.
26.	База даних	Створення бази даних співробітників відділу інформаційних технологій на основі зв'язаних списків
27.	Зовнішнє сортування	Розробка пакету програм з реалізації алгоритмів зовнішнього сортування (однофазне, двофазне, природне злиття)
28.	Розробка програми тестування знань.	Програма повинна передбачати модуль що створює тест, модуль навчання, модуль тестування та виведення результатів (наприклад порівняльний рейтинг у вигляді діаграм). Навчання та тестування повинне супроводжуватися графічними ілюстраціями. Тематика та уточнений склад програми виберіть самостійно.
29.	Розробка програми тестування знань.	Програма повинна передбачати модуль що створює тест, модуль навчання, модуль тестування та виведення результатів (наприклад порівняльний рейтинг у вигляді діаграм). Навчання та тестування повинне супроводжуватися графічними ілюстраціями.
30.	Побудова на площині графіків функцій	Типи графіків узгоджуються з керівником: полярні, параметричні, дискретні та ін. Програма повинна передбачати введення функцій, зміну графічного подання графіків (товщина, колір, стиль ліній) та зміну систем координат, збереження результатів, відображення декількох графіків, штриховка вказаних областей та ін.
31.	Розробка навчальних та демонстраційних програм	<ul style="list-style-type: none"> • демонстрація роботи основних керуючих структур в програмуванні (цикли, розгалуження, лінійні обчислення, рекурсія і т.п.) • програма моделювання черги, стека, стека з двома входами з графічним представленням даних. • демонстрація роботи рекурсії; • демонстрація графічних можливостей; • розробка інформаційної діалогової програми для допомоги абітурієнтам у виборі спеціальностей університету. • розробка програми презентації спеціальності "інформатика". • розробка навчальних програм з окремих розділів вищої математики.
32.	Деревовидні таблиці	Розробка пакету програм з реалізації обробки деревовидних таблиць

33.	Мінімальні остовні дерева	Розробка пакету програм з реалізації алгоритмів Прима та Крускала.
34.	Програмування обчислень спеціальних функцій і побудова їх графіків.	Вхідні дані: деякий клас спеціальних функцій та точність їх обчислення. Результат: таблиці значень цих функцій та їх графічне зображення для різних значень параметрів цих функцій. Передбачити збереження таблиць у вигляді текстових файлів, зміну графічного оформлення, масштабування графіків. Провести порівняння з відомими табличними даними. Вибір спеціальних функцій та метод їх обчислення виберіть самостійно (наприклад інтегральні функції, інтегральний синус і косинус, гамма функції, функції Беселя, функції Ейрі, інтеграли Френеля, Струве, Ангера і Вебера, повні еліптичні інтеграли).
35.	Розробка програми для розв'язання типових задач вищої математики.	Передбачити розвинену систему меню з вибору типу задач, введення початкових даних та формулювання умов задач (якщо необхідно то з поданням графічних зображень). Результати розв'язання задач подавати не тільки у вигляді числової відповіді, а у разі необхідності і з графічним зображенням та текстовими коментарями. Передбачити збереження умов та отриманих розв'язків. Типи задач вищої математики виберіть самостійно.
36.	Програмування розкладання функції в ряд Тейлора та дослідження збіжності ряду.	Вхідні дані: декілька функцій, точність розкладання. Результат: коефіцієнти розкладання. Передбачити чисельний та графічний аналіз точності розкладання (в залежності від кількості членів розкладання) та порівняння з точними значеннями. Передбачити можливість зміни графічного оформлення графіків та збереження отриманих результатів. Клас функцій для розкладання виберіть самостійно.
37.	Розробка програм аналізу та генерації псевдовипадкових чисел	Розробка програми генерації псевдовипадкових чисел в залежності від вибраної функції розподілення з можливістю роботи з файлами (запис та зчитування функції), а також результатів генерації.
38.	Переведення числа в різні системи числення (2-у, 8-у, 10-у, 16-у)	Розробка програми переведення числа в різні системи числення з можливістю роботи з файлами (запис та зчитування масиву чисел), а також результатів розв'язку.
39.	Учбові та навчаючі програми	Розробка мультимедійного пакету програм для підвищення ефективності засвоювання інформації студентами під час навчання.

40.	Задачі на графах	Розробка програми рішення пакету задач з використанням графів. Наявність доріг та відстань між містами задані орієнтованим графом. визначити: а) найкоротший шлях з А в Б; б) всі можливі шляхи з А в Б; в) мінімальні відстані від а до будь-якого з міст; г) всі міста, не пов'язані між собою дорогами
41.	Метод Гауса розв'язання систем лінійних алгебраїчних рівнянь (СЛАР)	Розробка програми розв'язання СЛАР з можливістю роботи з файлами матриць та результатів розв'язку.
42.	Обчислення зворотної матриці, визначника матриці	Розробка програми обчислення зворотної матриці, визначника матриці з можливістю роботи з файлами матриць та результатів розв'язку.
43.	Мультимедійний конспект по програмуванню на мові С (анімація + гіпертекст)	Гіпертекстовий документ, теоретичний, довідковий, практичний матеріал, що містить питання по програмуванню на мові С із запуском прикладів у вигляді файлів *.exe, анімацією.
44.	Наближення функцій методом найменших квадратів	Розробка програми наближення функцій з можливістю роботи з файлами (запис та зчитування функцій) та результатів розв'язку.
45.	Моделювання роботи аеропорту (розробка бази даних)	Програма-диспетчер, що управляє процесом зльоту і посадки літаків, реєструє аварійні ситуації, видає інформацію користувачу про проведені посадки, зльоти і стан польоту літаків.
46.	Моделювання роботи залізничного вокзалу (розробка бази даних)	Програма-диспетчер, що управляє процесом приходу та відправлення потягів, реєструє аварійні ситуації, видає інформацію користувачу про проведення посадки на потяги, стан потягів.
47.	Каталог літератури по сучасним мовам програмування	Багато віконна оболонка для відображення переліку, здійснення гнучкого пошуку, обробки для статистики і т. і.
48.	Програма моделювання виконання математичних операцій	Виконання векторних, матричних, арифметичних розрахунків, побудова графіків, діаграм та ін.
49.	Англо-російський (український) перекладач	Перекладач-підрядковий переклад з узгодженням відмінків в пропозиціях. Об'єм словника від 2000 слів. Здійснює переклад окремих слів або цілих фраз. Запрошуються режими висновку перекладу (нове вікно або в старому вікні).
50.	Программ перевірки правопису та граматики	Інтерактивна оболонка читання файлів, пошуку помилок в тексті і їх коректування та вибору зі словнику.

51.	Програма роботи з матрицями	Матричні перетворення: складання, перемноження, ділення, піднесення до степеню, пошук зворотної матриці та ін.
52.	Програма побудови графіків по введеним функціям	Інтерактивна багато віконна оболонка, яка демонструє вибір функції, введення початкових даних, вибір методу і виду побудови графіків.
53.	Електронний щоденник робочого дня, тижня	Інтерактивна багато віконна оболонка з можливостями ділового щоденника з відображенням переліку запланованих справ, виконаних заходів, ін. Передбачити можливість шифрування та архівування.
54.	Програма побудови розкладу занять на факультеті	Інтерактивна багато віконна оболонка побудови розкладу занять на факультеті з можливістю пріоритетів користувача та створення звітів в Excel.
55.	Множення матриць і операції, які з ними зв'язані	Алгоритм Штрассена для множення матриць. Обертання матриць

Робочу програму навчальної дисципліни (силабус):

Складено старшим викладачем Романовим Валерієм Володимировичем

Ухвалено кафедрою СП (протокол № 13 від 17.06.2024)

Погоджено Методичною комісією ІПСА (протокол №10 від 24.06.2024)