



ПРОГРАМУВАННЯ.

Частина 2. Програмування мовою С

Робоча програма кредитного модуля (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Системне програмування та спеціалізовані комп'ютерні системи</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>1 курс, весняний семестр</i>
Обсяг дисципліни	<i>5 кредитів ECTS</i>
Семестровий контроль/ контрольні заходи	<i>2 семестр: МКР, РГР, екзамен</i>
Розклад занять	<i>rozklad.kpi.ua</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор: <i>к.т.н., доц. Тарасенко-Клятченко О.В., easilly7@gmail.com</i> Практичні: <i>к.т.н., доц. Тарасенко-Клятченко О.В., easilly7@gmail.com</i> Лабораторні: <i>к.т.н., доц. Тарасенко-Клятченко О.В., easilly7@gmail.com</i> <i>ас. Кучмії О.О., yuo@ukr.net</i>
Розміщення курсу	<i>https://sites.google.com/view/otee-k</i>

Програма навчальної дисципліни

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

Навчальна дисципліна "Програмування" спрямована на вивчення теоретичних та методологічних основ побудови програм мовою програмування С з урахуванням сучасних концепцій і тенденцій розвитку, оволодіння інструментальними засобами створення таких програм, отримання практичних навичок розробки програмного забезпечення при вирішенні прикладних задач різного ступеня складності. Відповідна теоретична та практична підготовка формує базові навички з програмування і є основою для успішного опановування фахових дисциплін.

Метою дисципліни є формування у студентів здатності розробляти програмне забезпечення для вирішення прикладних задач різного ступеня складності мовою програмування С з використанням методологій процедурного і модульного програмування.

Предмет навчальної дисципліни – парадигма імперативного програмування, методології структурного, процедурного і модульного програмування, мова програмування С, інструментальні засоби створення програмного забезпечення мовою С.

Вивчення дисципліни «Програмування» сприяє формуванню у студентів наступних компетентностей.

Компетентності, формуванню яких сприяє дана дисципліна:

- ЗК 01 Здатність до абстрактного мислення, аналізу та синтезу*
- ЗК 07 Вміння виявляти, ставити та вирішувати проблеми*
- ЗК 08 Здатність працювати в команді*
- ФК 2 Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення*
- ФК 02 Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення*
- ФК 16 Здатність до алгоритмічного та логічного мислення*

Формування зазначених компетентностей забезпечує досягнення наступних програмних результатів навчання:

- ПРН 03 Знати новітні технології в галузі комп'ютерної інженерії*
- ПРН 04 Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті*
- ПРН 07 Уміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності*
- ПРН 08 Вміти системно мислити та застосовувати творчі здібності до формування нових ідей*
- ПРН 12 Вміти ефективно працювати як індивідуально, так і у складі команди*
- ПРН 16 Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення*
- ПРН 18 Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях*

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

Для успішного засвоєння дисципліни «Програмування» необхідно і достатньо мати математичну підготовку рівня середньої школи, базові знання основ програмування в обсязі, передбаченому програмою середньої школи. Даний курс підтримується курсом «Структури даних та алгоритми», який викладається паралельно в першому та другому семестрах. Це сприяє кращому засвоєнню та розумінню головних концепцій в основній схемі підготовки задач для розв'язку на ЕОМ.

Успішне оволодіння знаннями з даної дисципліни забезпечує вивчення надалі таких курсів, як «Системне програмування» (ПО 05), «Об'єктно-орієнтоване програмування» (ПО 09), «Паралельне програмування» (ПО 10).

3. Зміст навчальної дисципліни

Розділ 1. Рекурсія.

Тема 1.1. Рекурсивні обчислення. Рекурсивні функції. Виклик рекурсивних функцій. Глибина рекурсії. Форми рекурсії. Переваги і недоліки використання рекурсивних алгоритмів.

Розділ 2. Файли. Файлові потоки.

Тема 2.1. Файли. Логічний та фізичний файл. Файли та потоки. Структура файлу.

Тема 2.2. Загальні відомості про бібліотеки потокового вводу-виводу. Функції для обміну з потоками. Файли послідовного доступу. Створення, читання, запис даних. Буферизація.

Тема 2.3. Файли прямого (довільного) доступу. Створення, читання, запис даних.

Тема 2.4. Текстові файли.

Розділ 3.	Використання аргументів командного рядка. Багатофайлові проєкти.
Тема 3.1.	Аргументи командного рядка.
Тема 3.2.	Препроцесор. Заголовочні файли. Файли як модулі. Створення бібліотеки.
Розділ 4.	Вказівники
Тема 4.1.	Вказівники. Операції над вказівниками. Вказівники на масив, на структуру, на функцію. Передача функції як параметр в іншу функцію.
Розділ 5.	Динамічна пам'ять
Тема 5.1.	Динамічні програмні об'єкти. Динамічні змінні та масиви. Породження та знищення динамічних об'єктів.
Тема 5.2.	Динамічні структури даних: однонаправлені та двонаправлені списки. Кільцеві списки, стеки, черги, списки, бінарні дерева.
Розділ 6.	Об'єктно-орієнтоване програмування.
Тема 6.1.	Основні поняття ООП. Властивості та їх класифікація. Інкапсуляція та спадкування, правила спадкування. Клас, як абстрактний тип даних. Класи та об'єкти. Абстрактні методи. Класи та шаблони. Створення сімейства класів.
Тема 6.2.	Шаблони класів. Контейнерні класи. Множинне успадкування. Віртуальні базові класи. Об'єкти та модулі. Поліморфізм. Конструктори. Види конструкторів.. Динамічні об'єкти. Деструктори. Сумісність об'єктних типів.
Тема 6.3.	Вказівники на методи та делегування методів. Інтерфейси та делегування інтерфейсів.
Тема 6.4.	Додаткові відомості про функції

4. Навчальні матеріали та ресурси

Базовою літературою, обов'язковою для прочитання, є:

1. Глибовець, М. М. Мова програмування Сі : навч. посіб. з лабораторного практикуму / М. М. Глибовець, І. В. Кравченко, Є. С. Бубенчиков. – К.: НаУКМА, 1999. 22. Глибовець, М. М. Мова програмування Сі : навч.-метод. посіб. / М. М. Глибовець, В. І. Ляшко, В. С. Проценко. – К.: ВД "КМ Академія", 2002.
2. Дорошенко, А. Ю. Алгеброалгоритмічні основи програмування / А. Ю. Дорошенко, Г. С. Фінін, Г. О. Цейтлін. – К.: Наук. думка, 2004.
3. Енциклопедія кібернетики; т. 1-2. – К.: Наук. думка, 1973.
4. Проценко, В. С. Техніка програмування мовою Сі / В. С. Проценко, П. Й. Чаленко, А. Б. Ставровський. – К.: Либідь, 1993.

Додаткова література:

1. Браян В. Керніган, Деніс М.Річі. Мова програмування С. 232 с.
2. В.Ю. Вінник. Алгоритмічні мови та основи програмування: мова С. Житомир, ЖДТУ, 2007, 328 ст.
3. Войтенко В.В. та Морозов А.В. С/С++. Теорія та практика. Житомир, ЖДТУ, 2004, 324 ст.

Електронні ресурси:

1. <https://ela.kpi.ua/handle/123456789/31466>
2. <https://ela.kpi.ua/handle/123456789/40081>
3. <https://ela.kpi.ua/handle/123456789/45708>

Навчальний контент

4. Методика опанування навчальної дисципліни (освітнього компонента)

5.1 Лекційні заняття

Матеріали для вивчення дисципліни розміщені в електронному вигляді на сайті викладача (<https://sites.google.com/view/otee-k>) та в середовищі «Електронний кампус». Контент доступний студенту із будь-якого місця в мережі Інтернет.

Лекції по дисципліні проводяться із використанням сучасних мультимедійних презентаційних технологій.

№ з/п	Назва теми лекції та перелік основних питань
1	Рекурсія. Рекурсивні обчислення. Рекурсивні функції. Виклик рекурсивних функцій. Глибина рекурсії. Форми рекурсії. Переваги і недоліки використання рекурсивних алгоритмів.
2	Файли. Файлові потоки. Файли. Логічний та фізичний файл. Файли та потоки. Структура файлу.
3	Файли з послідовним доступом. Загальні відомості про бібліотеки потокового вводу-виводу. Функції для обміну з потоками. Файли послідовного доступу. Створення, читання, запис даних. Буферизація.
4	Файли з прямим доступом. Файли прямого (довільного) доступу. Створення, читання, запис даних.
5	Текстові файли. Текстові файли.
6	Використання аргументів командного рядка. Багатофайлові проекти. Аргументи командного рядка. Препроцесор. Заголовочні файли. Файли як модулі.
7	Використання аргументів командного рядка. Багатофайлові проекти. Створення бібліотеки.
8	Вказівники Вказівники. Операції над вказівниками.
9	Вказівники Вказівники на масив. Масиви вказівників.
10	Вказівники Вказівник на вказівник. Вказівник на структуру. Вказівник на функцію.
11	Динамічна пам'ять Динамічні програмні об'єкти. Динамічні змінні та масиви. Породження та знищення динамічних об'єктів.
12	Динамічна пам'ять Динамічні структури даних. Однонаправлені лінійні списки
13	Динамічна пам'ять Динамічні структури даних. Двонаправлені лінійні списки
14	Динамічна пам'ять

	<i>Циклічні списки</i>
15	Об'єктно-орієнтоване програмування. <i>Основні поняття ООП.</i>
16	Об'єктно-орієнтоване програмування. <i>Конструктори та деструктори</i>
17	Об'єктно-орієнтоване програмування. <i>Динамічні об'єкти. Деструктори. Сумісність об'єктних типів.</i>
18	Об'єктно-орієнтоване програмування. <i>Додаткові відомості про функції</i>

4.2 Практичні заняття

<i>№ з/п</i>	<i>Назва теми заняття та перелік основних питань</i>
1	<i>Розв'язок задачі з рекурсивними обчисленнями.</i>
2	<i>Використання файлів з послідовним доступом.</i>
3	<i>Використання файлів з прямим доступом.</i>
4	<i>Текстові файли.</i>
5	<i>Вказівники.</i>
6	<i>Вказівник на функцію. Передача функції параметром.</i>
7	<i>Динамічні дані. Динамічні масиви.</i>
8	<i>Лінійні списки.</i>
9	<i>Елементи ООП.</i>

На практичних заняттях розв'язуються задачі з використанням мови програмування C та середовища розробки програм (IDE), які надаються розробниками для навчальних цілей безкоштовно.

5.3. Лабораторні роботи

<i>№ з/п</i>	<i>Назва лабораторної роботи</i>	<i>Кількість ауд. годин</i>
1	<i>Файли</i>	<i>18</i>
2	<i>Наближене обчислення інтеграла</i>	<i>6</i>
3	<i>Побудова графіка функції</i>	<i>6</i>
4	<i>Класи і об'єкти</i>	<i>6</i>

Лабораторні заняття виконуються з використанням мови програмування C та версій середовищ розробки програм (IDE), які надаються розробниками для навчальних цілей безкоштовно. Під час їх проведення використовуються методичні вказівки до виконання лабораторних робіт по дисципліні, розміщені за посиланням <https://ela.kpi.ua/handle/123456789/40081>.

В умовах дистанційного навчання усі види занять, у тому числі контрольні заходи, можуть бути проведені з використанням сервісу Zoom.

5. Самостійна робота студента

До самостійної роботи студента відноситься виконання індивідуальних завдань з тематики, яка виноситься на лабораторні роботи, а також опрацювання теоретичного

матеріалу за наданими текстами лекцій та додатковою літературою, у тому числі за темами, які винесені на самостійне вивчення (згідно таблиці 1). Усі навчальні матеріали (презентації лекцій, практичних занять, методичні вказівки до виконання лабораторних робіт) розміщені в електронному вигляді в e-mail груп, а також в середовищі «Електронний кампус». Навчальний контент доступний із будь-якого місця в мережі Інтернет. На самостійну роботу студент має витратити кількість годин, що співмірна із кількістю годин, проведених ним на аудиторних заняттях.

Таблиця 1. Питання, які виносяться на самостійне опрацювання

№ з/п	Назва теми, що виносяться на самостійне опрацювання
1	Рекурсивні функції
2	Файлові потоки
3	Перевантаження функцій
4	Препроцесор та бібліотеки
5	Вказівники
6	ООП

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Студент має вивчати дисципліну протягом другого семестра, дотримуючись календарного плану виконання завдань лабораторних робіт, вивчення тем лекційного матеріалу, виконання практичних задач та виконання модульних контрольних робіт. Усі завдання студент має виконувати самостійно і вчасно.

Завдання з лабораторної роботи вважається виконаним, якщо студент захистив його у викладача (показав працездатність, відповів на усі питання) та розмістив звіт з виконання даної роботи на платформі листанційного навчання «Сікорський». Несвоєчасним вважається виконання завдання із затримкою більше 1 тижня. За несвоєчасну здачу лабораторних робіт передбачені штрафні бали. Такі обмеження стимулюють студента організувати систематичне виконання завдань та не допускати значного накопичення незданих лабораторних робіт на кінець семестру.

Для перевірки як теоретичних знань, так і практичних навичок в семестрі передбачена модульна контрольна робота (МКР), що складається з 4 частин в першому та 4 частин в другому семестрах, які проводяться під час практичних занять.

Оцінювання студентів здійснюється згідно рейтингової оцінки рівня підготовки студентів з дисципліни. Поточний стан успішності студенти можуть бачити в системі «Електронний кампус».

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

8.1 Поточний контроль

Поточний контроль результатів навчання передбачає виконання студентами лабораторних робіт, написання РГР та МКР.

Вагові бали кожного завдання лабораторної роботи залежать від його складності і варіюються у діапазоні від 0 до 5 балів відповідно до табл. 2.

Таблиця 2. Вагові бали та критерії оцінювання лабораторних робіт

1	Файли	3	9	2	1
2	Наближене обчислення інтеграла	1	2	1	1
3	Побудова графіка функції	1	2	1	1
4	Класи і об'єкти	1	2	1	1
Разом за лабораторні роботи		30 балів			

Критерії оцінювання лабораторних робіт включають якість розробки алгоритму програми, безпомилковість її виконання, якість захисту розробленої програми та підготовки звіту щодо виконаної роботи (див. табл. 2).

За несвоєчасну здачу лабораторних робіт передбачені штрафні бали (% зниження згідно до табл. 3).

Таблиця 3. Штрафи за затримку здачі індивідуальних завдань комп'ютерного практикуму

Термін затримки	Величина від'ємного бала
2 тижні	1
4 тижні	2
6 тижнів	3
>8 тижнів	4

МКР проводиться у вигляді письмової роботи або тестів, питання в яких можуть бути як теоретичними, так і практичними.

Вагові бали за МКР наведено в табл.4.

Таблиця 4. Вагові бали за МКР

№ та тема контрольної роботи	Ваговий бал
2 семестр	
Контрольна робота №1. Файли.	5
Контрольна робота №2. Вказівники.	5
Контрольна робота №3. Лінійні списки.	5
Контрольна робота №4. Підсумкова.	5
Разом	20

РГР передбачає виконання індивідуальних завдань згідно до встановлених варіантів. Метою РГР є поглиблення знань і напрацювання навичок роботи з різноманітними структурами даних, операторами, графічними можливостями. Ваговий бал РГР – 10 балів.

8.2 Календарний контроль

Проводиться двічі на семестр як моніторинг поточного стану виконання вимог робочої програми.

8.3 Семестровий контроль

Семестровий контроль результатів навчання проводиться у вигляді екзамену.

Умови допуску до семестрового контролю (екзамену):

- зарахування усіх лабораторних робіт,
- виконання і захист РГР
- стартовий рейтинг $rC \geq 30$ балів (не менше 50 % від RC).

Розмір шкали рейтингу $R = 100$ балів, розмір стартової шкали $RC = 60$ балів, розмір екзаменаційної шкали $RE = 40$ балів.

Екзаменаційна оцінка складається з оцінки за результатами написання і налагодження декількох частин модульної програми відповідно до поставленої задачі, виконання практичного завдання та відповіді на теоретичне питання.

Ваговий бал програми - 20 балів, практичне завдання оцінюється у 10 балів, теоретичне – у 10 балів.

Підсумкова оцінка формується за результатами оцінювання знань та навичок студента в семестрі та на екзамені за формулою: $R = RC + RE$.

Підсумкова оцінка переводиться до залікової оцінки згідно з таблицею:

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

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н., доц., доц. Тарасенко-Клятченко О.В.

Ухвалено кафедрою СПСКС (протокол № 11 від 12.06.2024 р.)

Погоджено Методичною комісією факультету ПМ (протокол № 12 від 21.06.2024 р.)