



Кам'янець-Подільський національний
університет імені Івана Огієнка
Фізико-математичний факультет
Кафедра комп'ютерних наук

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ ПРОГРАМУВАННЯ

1. Загальна інформація про курс. Викладачі

Іванюк Віталій Анатолійович, доктор технічних наук, доцент, завідувач кафедри комп'ютерних наук
Телефон: +38 (068) 746 33 28

E-mail: wivanyuk@kpmu.edu.ua

Профайл: <https://cs.kpmu.edu.ua/2019/11/04/ivaniuk-vitalij-anatolijovych/>

Сторінка курсу в MOODLE: <https://moodle.kpmu.edu.ua/course/view.php?id=19926>

Консультації: щочетверга – 16.00-18.00 (очно та онлайн)

Мова викладання: українська

2. Анотація до курсу

Навчальна дисципліна присвячена сучасним технологіям структурного програмування, її вивчення дозволить проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, розробляти програмні моделі предметних середовищ.

Вивчення навчальної дисципліни супроводжується вивченням мови програмування С. Розглядається синтаксис та семантика базових конструктивних елементів мови: лексем, виразів та операторів. Детально розглядаються всі різновиди типів даних, як простих (арифметичних і вказівникових), так і складених (масивів, структур, об'єднань). Значна увага приділяється програмуванню функцій, які є основними структурними компонентами, розглядаються засоби організації роботи з файлами та динамічними структурами. Велика увага приділяється питанням практичного програмування задач, пов'язаних із опрацюванням числової та текстової інформації.

Набуті компетентності та практичні навички будуть використовуватись в подальшому у вивченні інших навчальних дисциплін, таких як: Методи обчислень, Об'єктно-орієнтоване програмування, Операційні системи, Комп'ютерні мережі, Організація баз даних та знань, Проектування інформаційних систем, Веб-технології та веб-дизайн, Технології захисту інформації, Моделювання систем, Технології розподілених систем та паралельних обчислень, а також для виконання курсових та дипломної роботи.

3. Мета і завдання курсу

Метою вивчення навчальної дисципліни "Програмування" є отримання навичок проектування складних програм з використанням сучасних технологій структурного програмування, розширення кругозору щодо можливостей програмування, формування професійного відношення до створення програмного продукту, розуміння того факту, що тільки ретельно продуманий вибір необхідної структури даних для конкретної задачі може бути гарантією написання програми, що відповідає сучасним вимогам.

Компетентності:

ЗК 01 Здатність до абстрактного мислення, аналізу та синтезу.

ЗК 02 Здатність застосовувати знання у практичних ситуаціях.

ЗК 03 Знання та розуміння предметної області та розуміння професійної діяльності.

СК 03 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.

СК 08 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.

4. Результати навчання

ПРН 01 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 02 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.

ПРН 05 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 09 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.

5. Формат курсу

Очний курс з елементами дистанційного навчання в системі Moodle

6. Обсяг і ознаки курсу

Найменування показників	Характеристика навчальної дисципліни
Рік навчання	1
Семестр навчання	1
Кількість кредитів ЄКТС	4
Загальний обсяг годин	120
Кількість годин навчальних занять	48
Лекційні заняття	16
Практичні заняття	8
Семінарські заняття	0
Лабораторні заняття	24
Самостійна та індивідуальна робота	72
Форма підсумкового контролю	Залік

7. Пререквізити і кореквізити курсу

Дисциплін, які мають бути вивчені раніше або вивчатися паралельно: Дискретна математика, Алгоритми та структури даних.

8. Технічне й програмне забезпечення, обладнання

Для проведення лекцій необхідно мультимедійне забезпечення. При проведенні лабораторних робіт лабораторії із необхідним програмним забезпеченням.

Visual Studio Code, компілятор C.

9. Політика курсу

Увесь навчальний контент розміщено в модульному середовищі навчання К-ПНУ імені Івана Огієнка – moodle. Підготовка та виконання завдань лабораторних робіт і модульної контрольної роботи є обов'язковими для кожного здобувача вищої освіти.

Академічна доброчесність. Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел списування, втручання в роботу інших здобувачів вищої освіти становлять приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі здобувача вищої освіти є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.

Відвідання занять. Очікується, що всі здобувачі вищої освіти відвідають усі лекції та лабораторні заняття курсу. Здобувачі вищої освіти мають інформувати викладача про неможливість відвідати заняття. Виконання усіх лабораторних робіт є обов'язковим. Студенти зобов'язані дотримуватися термінів виконання усіх видів робіт, передбачених курсом. Форми поточного та підсумкового контролю. Поточний контроль реалізується на лабораторних заняттях.

Підсумковий контроль зі змістових модулів (допуск до екзамену) виставляється за результатами поточного контролю і модульних контрольних робіт.

10. Схеми курсу

Теоретична частина:

№	Назви змістових модулів і тем	Кількість годин
	Загалом	16
	Змістовий модуль 1. Організація програм	8
1	Поняття алгоритму. Мови програмування	2
2	Типи даних. Вирази та операції. Форматне виведення та введення даних	2
3	Оператори. Вказівники	2
4	Масиви. Символьні рядки	2
	Змістовий модуль 2. Структури, робота з пам'яттю та файлами	8
5	Структури та об'єднання	2
6	Функції	2
7	Класи пам'яті. Обмін даними з файлами	2
8	Робота з файлами. Директиви препроцесора	2

Практичні роботи:

№	Назви змістових модулів і тем	Кількість годин
	Загалом	8
	Змістовий модуль 1. Організація програм	4
1	Оператори. Вказівники	2
2	Масиви. Символьні рядки	2
	Змістовий модуль 2. Структури, робота з пам'яттю та файлами	4
3	Функції	2
4	Файли	2

Лабораторні роботи

№	Назви змістових модулів і тем	Кількість годин
	Загалом	24
	Змістовий модуль 1. Організація програм	14
1	Типи даних.	2
2	Вирази та операції	2
3	Оператор розгалуження	2
4	Циклічні алгоритми.	2
5	Адресна арифметика	2
6	Масиви.	2
7	Символьні рядки	2
	Змістовий модуль 2. Структури, робота з пам'яттю та файлами	10
8	Структури та об'єднання	2
9	Функції	2
10	Класи пам'яті даних	2
11	Робота з даними в динамічній пам'яті	2
12	Обмін даними з файлами	2

1	Директиви препроцесора	0
3		

Самостійна робота

№	Назви змістових модулів і тем	Кількість годин
	Загалом	72
	Змістовий модуль 1. Організація програм	36
1	Поняття алгоритму. Мови програмування	4
2	Типи даних. Вирази та операції	4
3	Форматне виведення та введення даних	4
4	Оператори	8
5	Вказівники	6
6	Масиви	10
	Змістовий модуль 2. Структури, робота з пам'яттю та файлами	36
7	Структури та об'єднання	6
8	Функції	8
9	Класи пам'яті даних	6
10	Робота з даними в динамічній пам'яті	6
11	Обмін даними з файлами	6
12	Директиви препроцесора	4

11. Система оцінювання та вимоги

Розподіл балів між поточним і модульним контролем подано у таблиці.

Загалом	100
ЗМ1	50
ЗМ2	50

Курс складається з 2-х навчальних (змістовних) модулів.

Поточний контроль полягає в перевірці теоретичних знань та практичних умінь і навичок під час лабораторних занять.

Результати перевірки самостійної роботи, в тому числі виконання домашніх завдань, студентів також входять до поточного контролю. Максимальний бал оцінки поточної успішності студентів на практичних заняттях приймається рівним 12.

Підсумковий контроль проводиться у формі заліку.

Рейтингова оцінка студентам виставляється відповідно до Положення про рейтингову систему оцінювання навчальних досягнень здобувачів вищої освіти Кам'янець-Подільського національного університету імені Івана Огієнка після проведення всіх навчальних занять та контрольних заходів з навчальної дисципліни.

Студенти, які мають академічну заборгованість за результатами підсумкового контролю, зобов'язані ліквідувати її в терміни, визначені графіком ліквідації академічної заборгованості.

12. Рекомендована література

Основна література:

1. Алексенко О. В. Технології програмування та створення програмних продуктів: конспект лекцій. Суми, Сумський державний університет, 2013. 133 с.
2. Пилипюк Т.М. Основи програмування. Кам'янець-Подільський, 2022. 92 с.
3. Пилипюк Тетяна, Мясковська Марина. Розробка програм засобами візуального програмування. Кам'янець-Подільський, 2023. 136 с.
4. Караванова Т.П. Основи алгоритмізації та програмування. 750 задач з рекомендаціями та прикладами: Посібник. Київ, Форум, 2002. 286 с.
5. Шпак З.Я. Програмування мовою C: Навчальний посібник. - Львів: Оріяна-Нова, 2006. - 432 с.

Додаткова література

6. Роберт С. Мартін. Чистий код: Створення і рефакторинг за допомогою agile. Фабула. 2019. 448 с.
7. Роберт С. Мартін. Чистий Agile: назад до основ. Фабула. 2021. 224 с.
8. Роберт С. Мартін. Чиста архітектура: Мистецтво розроблення програмного забезпечення. Фабула. 2020. 368 с.
9. Роберт С. Мартін. Чистий кодер: Кодекс поведінки для професійних розробників. Фабула. 2023.

256 c.