


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра оптоелектроніки та інформаційних технологій

Затверджено

На засіданні
кафедри оптоелектроніки та
інформаційних технологій
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № 6 від 29 08. 2021 р.)

Завідувач кафедри:


Олег КУШНІР

Силабус з навчальної дисципліни
“Основи програмування”,
що викладається в межах ОПП
“ Високопродуктивний комп'ютинг ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2021 р.

Назва дисципліни	Основи програмування
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79011
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій Кафедра оптоелектроніки та інформаційних технологій
Галузь знань, шифр та назва спеціальності	12 – Інформаційні технології 121 – Інженерія програмного забезпечення
Викладачі дисципліни	Крупич Олег Миколайович, доктор фіз.-мат. наук, ст.н.сп., професор; Катеринчук Іван Миколайович, канд. фіз.-мат. наук, доцент, доцент; Дуфанець Марта Василівна, канд. фіз.-мат. наук, доцент, доцент.
Контактна інформація викладачів	oleh.krpyuch@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/krpyuch-oleh-mykolayovych ; ivan.katerynychuk@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/katerynychuk-ivan-mykolajovych ; marta.dufanets@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/dufanets-marta-vasylivna
Консультації з питань навчання по дисципліні	Консультації в день проведення лекційних занять (за попередньою домовленістю): ауд. 128, корпус факультету електроніки та комп'ютерних технологій, вул. Ген. Тарнавського 107, м. Львів. Також можливі он-лайн консультації через MS Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://electronics.lnu.edu.ua/course/osnovy-prohramuvannia-ipz
Інформація про дисципліну	Дисципліна «Основи програмування» є нормативною дисципліною зі спеціальності 121 – Інженерія програмного забезпечення для освітньої програми «Високопродуктивний комп'ютинг», яка викладається в 1 семестрі в обсязі 5,0 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено таким чином, щоб ознайомити студентів з базовими поняттями про: основи програмування, абстракцію та алгоритми, компіляцію та інтерпретацію, середовище розробки, ключові слова, ідентифікатори та модифікатори, оператори, оператори галуження та цикли, функції, масиви одномірні та багатомірні, стрічки та роботу з ними, структуру пам'яті Сі-програми, вказівники, структури, систему вводу/виводу, роботу з файловою системою, директиви препроцесора.
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни «Основи програмування» є одержання студентами необхідних теоретичних та практичних знань з алгоритмізації та програмування, а також формування в студентів практичних навичок, які б дали змогу ефективно застосовувати знання з основ програмування в подальшій роботі та навчальному процесі.
Література для вивчення дисципліни	Основна література: 1. ISO/IEC 9899:2018 [ISO/IEC 9899:2018] Information technology — Programming languages — C 2. Брайан В. Керніган, Деніс М. Річі. Мова програмування C. Друге видання, 2015. – 232 с. http://programming.in.ua/programming/c-language/227-book-programming-c-kernighan.html

	<p>3. Хвищун І.О. Алгоритмізація та програмування: Лекції. Матеріали для самостійної роботи. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2017. –278с.</p> <p>Додаткова література:</p> <p>4. David M. Harris, Sarah L. Harris. Digital Design and Computer Architecture. Second Edition. Elsevier, Inc. -2013. -690 p.</p> <p>5. Шпак З.Я. Програмування мовою С. Навчальний посібник. – Львів: Оріяна-Нова, 2006. – 432 с.</p> <p>6. Вінник В.Ю. Алгоритмічні мови та основи програмування: мова С. Навчальний посібник. – Житомир: ЖДТУ, 2007. – 328 с.</p> <p>7. Daniel Zingaro. Algorithmic Thinking: A Problem-Based Introduction. No Starch Press. – 2020. – 408 p.</p> <p>8. Robert Seacord. Effective C: An Introduction to Professional C Programming. No Starch Press. – 2020. – 272 p.</p> <p>9. Anthony Wallit. Learning C programming: An Informative and In-depth Guide to C Programming. Independently published. – 2021. – 250 p.</p> <p>10. Jeff Szuhay. Learn C Programming: A beginner's guide to learning the most powerful and general-purpose programming language with ease, 2nd Edition. Packt Publishing. – 2021. – 742 p.</p>
<p>Обсяг курсу</p>	<p>Загальний обсяг: 150 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 год. лабораторних робіт. Самостійної роботи: 86 год.</p>
<p>Очікувані результати навчання</p>	<p>У результаті вивчення даного курсу студент буде:</p> <ul style="list-style-type: none"> – знати актуальність вивчення алгоритмів та мов програмування, методи представлення алгоритмів, способи програмної реалізації алгоритмів та засоби відлагодження програм, а також володіти здатністю до абстрактного мислення (в т.ч. алгоритмічного та логічного), аналізу та синтезу при створенні програмних продуктів; – вміти: класифікувати та розв’язувати складні спеціалізовані завдання інженерії програмного забезпечення, пов’язані зі створенням та програмною реалізацією алгоритмів, вміти застосовувати знання основ програмування в процесі практичного написання програмного коду, використовувати сучасні середовища розробки під час створення та відлагодження програм різного призначення. <p>Після вивчення даного курсу «Основи програмування» здобувачі набувають таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ФК15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>ФК23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи різні методології та засоби програмування з метою забезпечення їх високої надійності та продуктивності в роботі.</p> <p>ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об’єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p>

	<p>ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПРН08. Вміти розробляти людино-машинний інтерфейс.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.</p>
Ключові слова	Абстракція, алгоритм, компіляція, інтерпретація, середовище розробки, типи даних, ідентифікатор, оператор, цикл, функція, масив, стрічка, пам'ять, вказівник, структура, ввід, вивід, файл, директива.
Формат курсу	Очний.
Теми	Див. СХЕМА КУРСУ.
Підсумковий контроль, форма	Іспит в кінці семестру.
Пререквізити	Для вивчення курсу студенти потребують базових знань, одержаних при здобутті середньої освіти, насамперед з предметів «Інформатика», «Математика», «Фізика» та ін.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	<p>Для проведення лекційних занять: комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, відеокарта Nvidia GEFORCE GT1030 2048MB), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять: Комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, відеокарта Nvidia GEFORCE GT1030 2048MB). ОС Win10. MS Visual Studio 2017. Платформа Moodle та MS Teams, інтегроване середовище розробки Code::Blocks, онлайн компілятор мови програмування C «https://repl.it/languages/c»</p>
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться упродовж семестру за 100-бальною шкалою. Бали нараховуються за такими видами робіт з наступним співвідношенням:</p> <ul style="list-style-type: none"> • лабораторні роботи: 36% семестрової оцінки; максимальна кількість балів 36; • контрольні заміри (2 модулі): 14% семестрової оцінки; максимальна кількість балів 14; Контрольні заміри проводяться у формі тестових завдань. • іспит: 50% семестрової оцінки; максимальна кількість балів 50. <p>Загалом упродовж семестру 100 балів.</p> <p>Академічна доброчесність: Очікується, що лабораторні та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p>

	<p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового іспиту. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.</p>
Питання до екзамену	Перелік питань та завдань для проведення іспиту розміщаються на веб-сторінках курсу на платформах Moodle та MS Teams.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенні курсу.

**Схема курсу «Основи програмування»
для студентів спеціальності 121 – Інженерія програмного забезпечення
для освітньої програми «Високопродуктивний комп'ютинг»**

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	Вступ до курсу основи програмування. Історичні аспекти розвитку теоретичних та апаратних засобів для здійснення обчислень. Поняття про комп'ютер, інформацію, дані. Зміст етапів інформаційного процесу. Інформатика та комп'ютерні науки. Історія виникнення мови Сі. Місце Сі серед інших мов програмування.	Лекція	3, 4, 6		
2	Абстракція та алгоритми. Поняття абстракції. Двійкова/шістнадцяткова системи числення. ASCII таблиця. Принципи побудови зображення. Поняття алгоритму та його ефективності. Алгоритм половинного ділення. Способи представлення алгоритмів. Псевдокод.	Лекція	3, 4, 6, 7	Основи алгоритмізації. Найпростіші алгоритми. 2 год.	3 тиж. семестру
3	Компіляція та інтерпретація. Середовище розробки. Трансляція програм. Компілятори та інтерпретатори. Принципи побудови та структура проекту. Середовища розробки Scratch, Visual Studio, CodeBlock. Онлайн-розробка. Засоби відлагодження коду.	Лекція	3, 4, 6	Графічна мова програмування Scratch. 2 год.	4 тиж. семестру

4	Мова програмування Сі. Ключові слова, ідентифікатори та модифікатори. Складові частини проекту на Сі. Оператори і операнди. Змінні, оголошення змінних. Ідентифікатори. Ключові слова. Типи даних. Модифікатори типів. Модифікатори доступу. Локальні і глобальні змінні, область видимості змінної. Специфікатори зберігання.	Лекція	1, 2, 5, 6, 9, 10	Мова програмування Сі, компіляція, функції вводу/виводу. 2 год.	5 тиж. семестру
5, 6	Оператори. Оператор присвоювання та ініціалізація змінних. Перетворення типів при присвоюванні. Константи, символічні константи з зворотним слешем. Арифметичні оператори. Оператори відношення та логічні оператори. Побітові оператори. Оператор ?. Оператори & та *. Оператор sizeof. Оператор ,, Оператори . та ->. Оператори [] та (). Пріоритети операторів. Приведення типів компілятором. Примусове перетворення типів. Пробіли і дужки. Скорочення операторів.	Лекція	1, 2, 5, 8, 9, 10	Основні типи змінних. Операції та їх пріоритет. 4 год.	7 тиж. семестру
7	Оператори галузження та цикли. Поняття галузження за умовою, оператори: if і switch. Цикли: while, for і do/while. Оператори переходу: break, continue. Конструкція if-else-if. Розв'язок рівняння алгоритмом половинного ділення.	Лекція	1, 2, 5, 6, 8, 10	Умовні оператори. 2 год. Цикли. 2 год.	9 тиж. семестру
8	Функції. Функції, призначення, оголошення. Попереднє оголошення функції. Оголошення функції через прототип. Прототип функції з невідомим числом параметрів та без параметрів. Правила видимості функцій. Аргументи функцій та передача аргументу через значення або вказівник. Рекурсія. Вказівник на функцію.	Лекція	1, 2, 6, 8, 10	Функції. 4 год	10 тиж. семестру
9,10	Масиви, стрічки та робота з ними. Одномірні масиви. Багатомірні масиви. Передача масивів в функцію. Алгоритми сортування масивів Стрічки та стрічкові константи. Функції для роботи зі стрічками. Масиви стрічок. Індексвання з допомогою вказівників. Динамічні масиви. Ініціалізація розмірного та безрозмірного масиву.	Лекція	1, 2, 5, 8, 9, 10	Масиви. 2 год. Стрічки 2 год.	12 тиж. семестру
11, 12	Структура пам'яті Сі програми. Вказівники. Структура пам'яті Сі програми. Стек. Купа. Динамічне виділення пам'яті і вказівники. Вказівники. Вказівники на константи. Вказівники і масиви. Вказівники на символічні масиви. Масиви вказівників. Вказівники на вказівники. Вказівники на функції.	Лекція	1, 2, 5, 8, 9	Вказівники. 4 год.	13 тиж. семестру
13	Структури. Поняття про структури. Доступ до полів структури. Присвоєння структур. Масиви структур. Передача полів структури в функцію. Вказівники на структури. Масиви і структури в структурах. Вкладені структури. Бітові поля. Об'єднання. Переліки.	Лекція	1, 2, 5, 6, 8, 10	Структури. 2 год.	14 тиж. семестру
14	Система вводу / виводу. Оператор typedef. Система вводу / виводу. Текстові потоки. Двійкові потоки. Файли робота з файлами в Сі. Консольний ввід/вивід.	Лекція	1, 2, 6, 8, 10	Мова програмування Сі, компіляція, функції вводу/виводу. 2 год.	15 тиж. семестру
15	Робота з файловою системою. Функції файлової системи ANSI C. Вказівник на файл. Відкриття файлу. Читання і запис файлу. Закриття файлу.	Лекція	1, 2, 5, 6, 8, 10	Робота з файлами. 2 год.	16 тиж. семестру

16	Директиви. Директиви препроцесора, директиви #define та #undef. Директиви умовної компіляції. Директиви препроцесора, директиви #include. Оператор defined. Наперед визначені макроси Si..	Лекція	1, 2, 5, 8, 9		
----	--	--------	---------------	--	--