

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

Затверджено

На засіданні кафедри радіофізики та
комп'ютерних технологій
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № ____ від _____ червня 2021 р.)

Завідувач кафедри _____

Силабус з навчальної дисципліни
«Алгоритмізація та програмування»,
що викладається в межах ОПП «Комп'ютерні науки» першого
(бакалаврського) рівня вищої освіти для здобувачів зі
спеціальності
122 «Комп'ютерні науки»

Назва дисципліни	Алгоритмізація та програмування
Адреса викладання дисципліни	м. Львів, вул. Тарнавського, 107
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра радіофізики і комп'ютерних технологій
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології, 122 Комп'ютерні науки
Викладачі дисципліни	Хвищун Іван Олександрович , кандидат технічних наук, доцент
Контактна інформація викладачів	Ivan.Khvyshchun@lnu.edu.ua , xiocompan@gmail.com
Консультації з питань навчання по дисципліні відбуваються	Консультації проводяться в день проведення лекційних занять (за попередньою домовленістю) згідно розкладу. В режимі он-лайн: консультації проводяться за домовленістю з студентами чи групою студентів на платформі MS Teams та на інших електронних ресурсах, за попереднім погодженням часу (через електронну пошту викладача або телефонний дзвінок).
Сторінка дисципліни	
Інформація про дисципліну	Дисципліна «Алгоритмізація та програмування» є нормативною дисципліною циклу професійної та практичної підготовки зі спеціальності 122 Комп'ютерні науки для освітньої програми «Комп'ютерні науки», яка викладається в 1 семестрі в обсязі 4-х кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Сучасний фахівець, який працює у сфері комп'ютерних технологій, повинен володіти фундаментальними знаннями в області алгоритмізації та програмування. Під цим ми розуміємо системний підхід до розв'язання інформаційних задач, алгоритмічне мислення, знання термінології і сучасних засобів розробки та налагодження програмного забезпечення.
Мета та цілі дисципліни	Мета: одержання студентами базових знань стосовно сучасних методологій розробки програмного забезпечення в імперативно-модульній парадигмі програмування, зокрема – вивчення основ програмування алгоритмічними мовами Delphi та Visual C++, вивчення типів даних та алгоритмічних конструкцій цих мов, освоєння особливостей обробки даних з використанням арифметики з плаваючою (рухомою) комою, а також освоєння базових чисельних методів, що використовуються при розв'язанні задач, які виникають у природничих науках. Цілі: навчити студентів алгоритмічному мисленню, методології розробки програм: низхідної та висхідної, розуміти синтаксичні та семантичні конструкції алгоритмічних мов Delphi та C++, отримати навички розробки та налагодження програм у консольних та віконних режимах роботи програмних середовищ Embarcadero RAD Studio та Visual Studio 2019, а також, опанувати принципи розробки алгоритмів та програм для розв'язання найпростіших задач фізики та електроніки.
Література для вивчення дисципліни	Основна література: 1. <i>Іван Хвищун</i> . Алгоритмізація та програмування. Лекції. Матеріали для самостійної роботи. – Львів: Вид. центр ЛНУ ім. І. Франка, 2017. – 282 с. 2. <i>Хвищун І.О.</i> Програмування і математичне моделювання: Підруч. – К.: Видавничий дім “Ін Юре”, 2007. – 544 с. 3. <i>Marco Cantu</i> . Delphi 2010 Handbook, Piacenza (Italy), 2010. – 318 p. 4. <i>Ковалюк Т.В.</i> Основи програмування. К.: ВНУ Києв, 2005. – 400 с

	<p>5. <i>Осипов Д.</i> Delphi. Профессиональное программирование. – СПб.: Символ-Плюс, 2006. – 1056 с.</p> <p>6. <i>Осипов Д. Л.</i> Delphi. Программирование для Windows, OS X, iOS и Android. — СПб.: БХВ-Петербург, 2014. – 464 с.</p> <p>7. <i>Павловская Т.А.</i> C/C++. Программирование на языке высокого уровня.– СПб.: Питер, 2007. – 461 с.</p> <p>8. <i>Пахомов Б.И.</i> C/C++ и MS Visual C++ 2012 для начинающих. – СПб.: БХВ-Петербург, 2013. – 512 с.</p> <p>9. <i>Степаненко О.Е.</i> Visual C++.NET. Классика программирования. – М: Научная книга, К.; Букинист, 2010. – 768 с.</p> <p>10. <i>Ivor Horton's Beginning Visual C++2010.</i> Wiley Publish., Inc, 2010 – 1231p.</p>
Обсяг курсу	64 години аудиторних занять. З них 32 години лекцій, 32 години лабораторних робіт на кожну із 10 бригад та 56 годин самостійної роботи
Очікувані результати навчання	<p>В результаті вивчення даного курсу студент повинен:</p> <p>знати:</p> <ul style="list-style-type: none"> • базові алгоритми розв'язання простих задач та форми їхнього представлення; • правила роботи в середовищі сучасних операційних систем та їхні файлові системи; • принципи роботи в інтегрованих середовищах Embarcadero RAD Studio Delphi та Microsoft Visual C++ 2019; • структури та складові частини програм; • скалярні типи даних мов програмування Delphi та C++; • структуровані типи даних: масиви, записи, структури, модулі, файли; • основні оператори названих мов: присвоєння, безумовні переходи, умовні переходи, цикли, виклику підпрограм; • правила опису та використання підпрограм; • основні принципи, алгоритми та методи обчислювальної математики; <p>вміти:</p> <ul style="list-style-type: none"> • працювати із файловими системами персонального комп'ютера; • інсталювати та налаштовувати середовища Embarcadero RAD Studio та Microsoft Visual Studio; • працювати у консольних та віконних режимах у середовищах Delphi та Visual C++; • програмувати мовами Delphi та Visual C++ алгоритми обробки та відображення текстових, числових та графічних даних; • створювати алгоритми розв'язання простих задач, які належать до сфери застосування комп'ютерної техніки у науково-технічній сфері; • налагоджувати програми, використовуючи стандартні засоби налагодження названих середовищ програмування.
Ключові слова	Програмування. Алгоритм. Мова програмування. Середовище програмування. Тип даних. Оператор. Операнд. Операція. Файл.
Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Два колоквиуми впродовж семестру у письмовій формі та іспит в кінці семестру в усній та практичній формі

Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Математичний аналіз», «Алгебра та геометрія».
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Лекції на основі презентацій, лабораторні роботи на основі методичок та їхній захист на основі індивідуальних звітів, виконання індивідуальних завдань та їхній захист.
Необхідне обладнання	Пристрої мультимедіа. ОС Windows 10, Embarcadero RAD Studio 10, Visual Studio 2019, платформи MS Teams, Moodle,
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 + індивідуальні завдання: 40% семестрової оцінки; максимальна кількість балів 40. • контрольні заміри (2 модулі): 10% семестрової оцінки; максимальна кількість балів 10. • іспит: 50% семестрової оцінки, максимальна кількість балів - 50 <p>Загалом, упродовж семестру: 100 балів.</p> <p>Контрольні заміри проводяться у формі двох письмових колоквіумів та поточного захисту індивідуальних звітів по кожній лабораторній та індивідуальній роботі.</p> <p>Академічна доброчесність: Очікується, що лабораторні індивідуальні роботи студентів будуть їхніми оригінальними розробками та дослідженнями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної не доброчесності. Виявлення ознак академічної не доброчесності в роботі студента є підставою для її не зарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та виконають усі основні та додаткові завдання до кожної лабораторної роботи. Студенти повинні інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.</p> <p>Література. Основна та додаткова література надається викладачем в електронній формі виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до пошуку та використання іншої літератури та джерел по тематиці курсу, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали набрані при захисті лабораторних та індивідуальних робіт, бали за колоквіуми та за іспит. 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.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</p>
Питання до контрольних робіт	Питання до колоквіумів студенти отримують за тиждень до дати колоквіуму. Білети для підсумкового іспиту студенти отримують на останній лекції курсу.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Тиждень	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	<p>ІСТОРИЧНІ АСПЕКТИ ТА ОСНОВНІ ПОНЯТТЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ.</p> <p>Мета та задачі курсу. Правила виконання лабораторного практикуму. Принципи та способи поточного і підсумкового контролю успішності. Література основна та допоміжна. Моменти історії розвитку обчислювальної техніки: Ера механічних обчислювальних пристроїв. Ера електронно-обчислювальних пристроїв. Мікропроцесорна революція. Закон Мура. Коротка історія мов програмування. Сутність програмування. Основні терміни програмування. Компілятори, інтерпретатори, лінкери, завантажувачі. Етапи розроблення програмного забезпечення. Вимоги до програмування. Стиль програмування.</p>	Лекція	1, 2, 3, 4, 5, 6	.Ознайомлення з середовищем Delphi (Lazarus). Програмування навчального консольного та віконного проєктів: “Розв’язання квадратного рівняння”.	1 тиждень семестру
2	<p>МОВА ПРОГРАМУВАННЯ PASCAL (DELPHI).</p> <p>Алфавіт мови Pascal. Лексеми мови програмування. Зарезервовані (ключові) слова. Елементарні конструкції. Структура Pascal-програми. Поняття типів даних. Константи: типізовані та нетипізовані. Підпрограми, які працюють з порядковими типами. Порівняння типів даних Borland Pascal і Delphi. Цілочисельні типи даних. Дійсні (дробові) типи даних. Символьні типи даних. Рядкові типи даних. Вказівники: типовані і нетиповані. Текстовий тип. Логічні (булівські) типи даних. Діапазонний та перераховувальний типи даних. Вирази. Бітова арифметика. Поняття інтегрованого середовища розробки програм.</p>	Лекція	1, 2, 3, 4, 5, 6	Програмування консольного проєкту: “Трикутник і точка в Декартовій системі координат”	2 тиждень семестру
3	<p>ОПЕРАТОРИ. АЛГОРИТМИ РОЗВ’ЯЗАННЯ НЕЛІНІЙНИХ РІВНЯНЬ.</p> <p>Керуючі структури програми. Оператори мови Pascal (Delphi). Оператор присвоювання :=. Оператори безумовного переходу. Оператор GOTO. Оператор виклику процедури. Умовні оператори (розгалуження). Скорочена форма оператора If. Повна форма оператора If. Вкладені умовні оператори. Умовний оператор Case (оператор вибору). Приклад – програма-калькулятор. Оператори циклів. For – цикл із відомою кількістю повторень (з параметром). Цикл із передумовою While (доки). Цикл із після умовою Repeat (доки не). Процедури дочасного</p>	Лекція	1, 2, 3, 4, 5, 6	Програмування консольних проєктів: “Метод ділення навпіл (МДН)” та “Метод Ньютона (МН)”	3 тиждень семестру

	завершення циклу. Оператори введення і виведення. Алгоритми розв'язання нелінійних рівнянь. Метод ділення напіл (МДН). Особливі випадки МДН. Алгоритм МДН. Метод послідовних наближень (Ньютона) – теорія. Алгоритм методу Ньютона. Приклад програмної реалізації алгоритму методу Ньютона. Чисельне знаходження першої та другої похідних.				
4	ПІДПРОГРАМИ. АЛГОРИТМИ. Поняття підпрограми. Стандартні підпрограми. Процедури. Синтаксис опису процедур. Процедури з параметрами. Параметри вхідні та вихідні. Локальні та глобальні змінні процедури. Функції. Синтаксис опису функцій. Функції перетворення типів даних. Арифметичні функції. Процедури та функції роботи з порядковими типами даних. Рядкові процедури та функції. Основні стандартні функції Delphi. Поняття алгоритму. Властивості алгоритмів. Форми представлення алгоритмів. Порівняння форм запису алгоритмів. Приклад розв'язання квадратного рівняння. Постановка задачі. Описова форма представлення алгоритму. Графічна форма представлення алгоритму. Рекурсія. Приклади програм з рекурсією.	Лекція	1, 2, 3, 4, 5, 6	Програмування віконного проекту: “МДН+МН”.	4 тиждень семестру
5	МАСИВИ. РОЗВ'ЯЗУВАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРИЧНИХ РІВНЯНЬ (СЛАР). Масиви (Аггау). Вимірність масивів: одновимірні (вектори), двовимірні (матриці), тривимірні (тензори). Два способи опису масивів. Заповнення елементів масиву початковими значеннями. Введення масивів з клавіатури. Використання елементів масиву у виразах. Наочне відображення масивів на екрані або їхнє друкування. Стандартні функції для обробки масивів. Проект Delphi та його файли. Вікно екранного редактора для набору програми. Послідовність створення віконних проектів у Delphi. Приклад розробки віконного проекту. Розв'язування систем лінійних алгебричних рівнянь (СЛАР). СЛАР з двома невідомими. Метод Крамера розв'язування СЛАР. Теорія методу Гауса розв'язування СЛАР. Математичний аспект методу Гауса. Тестова задача. Алгоритм методу Гауса та його процедура. Алгоритм розв'язання СЛАР на основі методу LU-розкладення матриці. Алгоритм знаходження оберненої матриці.		1, 2, 3, 4, 5, 6	Програмування консольного проекту: “Метод Крамера”	5 тиждень семестру
6	МОДУЛІ. ЗАПИСИ. СИСТЕМИ НЕЛІНІЙНИХ РІВНЯНЬ (СНР). Модулі (Unit). Структура модуля.	Лекція	1, 2, 3, 4, 5, 6	Програмування віконного проекту: “Метод Гауса 3	6 тиждень семестру

	Інтерфейсна секція (Interface). Секція реалізації (Implementation). Секція початкової ініціалізації. Секція кінцевої фіналізації. Використання модулів. Записи (Record). Оператор приєднання With. Теорія методу Ньютона для розв'язання СНР. Алгоритм розв'язання СНР методом Ньютона. Блок-схема алгоритму Ньютона. Реалізація алгоритму методу Ньютона.			використанням компоненти StringGrid”.	
7	ГРАФІЧНІ МОЖЛИВОСТІ DELPHI. Основні поняття комп'ютерної графіки. Графічні можливості середовища Delphi. Кольори та стилі графічних об'єктів. Малювання ліній. Мольберт (Canvas). Олівець і пензлик. Виведення тексту. Методи викреслювання графічних примітивів. Виведення малюнків. Побудова графіків функцій, які задані таблицею чи аналітично. Математичний аспект: табулювання функції із записуванням результатів у масиви. Знаходження мінімального та максимального значення елементів масиву. Принципи та формули масштабування табличного представлення функцій перед виведенням їх на графічний екран. Формули коефіцієнтів масштабування та їхнє обчислення. Алгоритм практичної реалізації універсального графіка. Побудова рухомих осей координат, масштабної ґратки і підписування її вузлів реальними числовими значеннями. Побудова графіків на основі стандартної компоненти TChart.	Лекція	1, 2, 3, 4, 5, 6	Програмування проекту: “Універсальний графік на Canvas форми”.	7 тиждень семестру
8	ВКАЗІВНИКИ. ФАЙЛИ. Вказівники і динамічна пам'ять. творення динамічних масивів. Поняття файлу. Файли у Delphi. Описування файлів. Відкривання/закривання файлів. Стандартні процедури та функції для роботи з файлами. Робота з каталогами. Атрибути файлів. Робота з типізованими, нетипізованими та текстовими файлами. Переваги та особливості текстових файлів. Підпрограми для роботи з текстовими файлами. Приклад роботи з файлами. Використання можливостей VCL Delphi для роботи з файлами.	Лекція	1, 2, 3, 4, 5, 6	Програмування віконного проекту: “Універсальний графік на TChart”	8 тиждень семестру
9	ВСТУП У ПРОГРАМУВАННЯ МОВОЮ Сі/C++. Основні терміни і поняття мов програмування. Порівняння синтаксису мов С++ та Delphi. Алфавіт, символи та ключові слова мови С++. Правила побудови ідентифікаторів. Коментарі. Типи даних: прості та структуровані. Класифікація типів даних у стандарті мови С++. Порівняльна таблиця стандартних типів даних. Порівняння скалярних типів даних Delphi та С++. Синтаксис	Лекція	7, 8, 9, 10	Освоєння середовища Visual С++. Навчальна програма розв'язання квадратного рівняння мовою С++ - консольний та віконний проекти.	9 тиждень семестру

	описування змінних. Константи та правила їхнього запису. Основні операції мови C++: унарні, бінарні, тернарна та їхня детальна характеристика. Приоритет операцій. Перетворення типів даних у C++. Оператори введення/виведення на текстовий екран: за правилами Cі, за правилами C++. Керуючі символи для виведення даних різних типів.				
10	ОПЕРАТОРИ МОВИ C++. ВИЗНАЧЕНИЙ ІНТЕГРАЛ. Оператор присвоєння. Оператори безумовного переходу: оператор goto <мітка>; , оператор виклику функції типу void. Порожній оператор “;”. Умовний оператор if, його скорочена і повна форма. Вкладені оператори if. Тернарний умовний оператор. Умовний оператор switch. Приклад – програма-калькулятор. Цикли. for – цикл із відомою кількістю повторень. Особливості оператора for в C++. while – ітераційний цикл з передумовою. do while – ітераційний цикл з післяумовою. Вкладені цикли. Оператори дочасного завершення циклу. Відмінності у роботі операторів циклів у порівнянні із їхніми аналогами в Delphi. Обчислення визначених інтегралів. Площа криволінійної фігури. Визначений інтеграл та його геометрична інтерпретація. Методи чисельного інтегрування. Методи прямокутників, трапецій та парабол (Сімпсона). Програмування алгоритмів обчислення визначених інтегралів. Обчислення інтегралів із заданою точністю. Алгоритм методу Монте-Карло.	Лекція	7, 8, 9, 10	Програмування віконного проекту: “Метод ділення навпіл та метод Ньютона” мовою C++.	10 тижень семестру
11	ФУНКЦІЇ У МОВІ C++. Формат опису функцій у мові C++. Формати опису функцій типу void та опису звичайних функцій. Оператор return. Приклади опису функцій. Головна функція main(). Прототипи функцій. Функції з аргументами і без аргументів. Загальна структура програми мовою C++. Обмін даними між функціями. Поняття стеку. Константні параметри. Повернення значення результату роботи функції. Рекурсивні функції. Деякі стандартні функції. Функції консольного введення/виведення даних у стилі Cі. Функції введення/виведення даних у стилі C++. Виведення повідомлень та значень. Специфікатори перетворення. Форматоване введення значень.	Лекція	7, 8, 9, 10	Програмування функцій для знаходження визначених інтегралів.	11 тижень семестру
12	ВКАЗІВНИКИ. ПОСИЛАННЯ. ДИНАМІЧНІ СТРУКТУРИ ДАНИХ. Вказівники (pointers): на об’єкт, на функцію, на void. Використання вказівників. Способи виділення та вивільнення динамічної пам’яті.	Лекція	7, 8, 9, 10	Програма обчислення інтегралів із заданою точністю	12 тижень семестру

	Операції з вказівниками. Посилання (references). Правила використання посилань. Вказівник на вказівник. Динамічні структури даних. Лінійні списки. Приклад однонапрявленого лінійного списку. Стек. Черга.			мовою C++.	
13	МАСИВИ. СТРУКТУРИ. ОБ'ЄДНАННЯ. Масиви. Одновимірні масиви (вектори). Двовимірні масиви (матриці). Синтаксис опису масивів одновимірних та багатовимірних. Модифікатори. Ініціалізація елементів початковими значеннями. Динамічні масиви. Виділення і вивільнення пам'яті під динамічні масиви. Доступ до елементів масиву. Рядки. Можливості бібліотечного класу string. Використання масивів у ролі параметрів функцій. Структури (struct). Формати опису структур. Доступ до полів структури. Ініціалізація структур. Об'єднання (union). Довжина об'єднання. Призначення об'єднань. Обмеження у використанні об'єднання. Визначення власних типів typedef. Перераховувальний тип (enum), особливості його використання.	Лекція	7, 8, 9, 10	Програмування методу LU-розкладення мовою C++.	13 тиждень семестру
14	ГРАФІЧНІ МОЖЛИВОСТІ VC++. РЯД ФУР'Є Елементи комп'ютерної графіки. Виведення графіки за допомогою GDI+. Поняття класу і простору імен. Класи і структури простору імен System.Drawing. Графічні контексти і графічні об'єкти. Основні простори імен для базових класів GDI+. Складові частини програми побудови графіка функції: під'єднання просторів імен до проекту; опис масивів; опис функції; табулювання функції із занесенням результатів у масиви; формули для обчислення коефіцієнтів масштабування; створення об'єкту для роботи з графікою; нанесення рухомих осей координат; виведення масштабованого графіка функції. Поняття спектру. Спектральний аналіз періодичних сигналів. Алгоритм побудови і табулювання ряду Фур'є. Функція, що будує і табулює ряд Фур'є. Графічне відображення спектрального аналізу періодичних функцій. Застосування рядів Фур'є в електроніці.	Лекція	7, 8, 9, 10	Програмування ряду Фур'є із графікою.	14 тиждень семестру
15	ФАЙЛИ У МОВІ СІ/C++. Робота з файлами у стилі мови Сі. Специфікатори режиму роботи з файлами у Сі. Обробка текстових файлів по символно. Обробка текстових файлів по рядках. Робота з файловими потоками у стилі мови C++. Робота з текстовими файловими потоками. Робота з бінарними файловими потоками.	Лекція	7, 8, 9, 10	Дослідження зв'язаних списків	15 тиждень семестру

16	ПЛАТФОРМА .NET. КЕРОВАНІ ДАНІ ТА КОД. Загальномовне середовище виконання CLR платформи .NET Framework. Версії мови C++/NET і C++/CLI. Посилальні типи даних і типи-значення. Використання стеку, некерованої і керованої куп у мові C++/CLI. Огляд сучасних технологій та парадигм програмування.	Лекція	7, 8, 9, 10	Робота з керованим кодом у режимі роботи C++/CLI.	16 тиждень семестру
----	--	--------	-------------	---	---------------------