



Програмування та алгоритмічні мови.

Частина 2. Програмування

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	124 «Системний аналіз»
Освітня програма	«Системний аналіз і управління»
Статус дисципліни	Нормативна
Форма навчання	очна(денна)
Рік підготовки, семестр	I курс, весняний семестр
Обсяг дисципліни	5 кредитів(150 годин)
Семестровий контроль/ контрольні заходи	Залік
Розклад занять	http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: старший викладач, Назарчук Ірина Василівна, inazarc@ukr.net , Комп'ютерний практикум: ст.викладач Назарчук Ірина Василівна, inazarc@ukr.net , ст.викладач Савченко Ілля Олександрович, savil.ua@gmail.com , доцент Просянкін-Жарова Тетяна Іванівна t.puman@gmail.com
Розміщення курсу	Googleclassroom

Програма навчальної дисципліни

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

Курс «Програмування та алгоритмічні мови» входить до циклу професійної підготовки обов'язкових освітніх компонентів освітньої програми. Кредитний модуль «Програмування та алгоритмічні мови. Частина 2. Програмування» є другим із трьох кредитних модулів курсу.

Предмет кредитного модуля – базові принципи об'єктно-орієнтованого програмування, основні прийоми створення закінченого програмного продукту в C++ з використанням об'єктно-орієнтованого підходу.

Метою навчальної дисципліни є формування у студентів наступних загальних здатностей.

- Здатність застосовувати знання в практичних ситуаціях (ЗК 1);
- Здатність планувати і управляти часом (ЗК 2);
- Здатність бути критичним і самокритичним (ЗК 8);
- Здатність до адаптації та дії в новій ситуації (ЗК 9);
- Здатність працювати в команді та автономно виконувати командні рішення (ЗК 12);
- Здатність забезпечувати та оцінювати якість виконуваних робіт (ЗК 14);
- Здатність реалізовувати свої права і обов'язки як члена суспільства, дотримуватися академічної доброчесності (ЗК 15).

За результатами вивчення дисципліни студент має бути здатним вирішувати професійні завдання та володіти такими компетентностями.

- Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних (ФК 6);

- Здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем, а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань (ФК 7);
- Здатність організувати роботу з аналізу та проектування складних систем, створення відповідних інформаційних технологій та програмного забезпечення (ФК 8).

Згідно з вимогами освітньої програми навчальної дисципліни після засвоєння кредитного модуля студенти мають продемонструвати наступні результати навчання.

- Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій (ПР08)
- Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень (ПР09)
- Розуміти і реалізовувати свої права і обов'язки як члена суспільства, усвідомлювати цінності вільного демократичного суспільства, верховенства права, прав і свобод людини і громадянина в Україні і дотримуватися академічної доброчесності (ПР16).

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

Міждисциплінарні зв'язки: забезпечується дисциплінами ПО 3 «Дискретна математика», ПО9 «Алгоритми і структури даних», ПО 19 «Математична логіка і теорія алгоритмів», забезпечує дисципліни ПО 20 «Об'єктно-орієнтоване програмування», ПО 13 «Організація баз даних та знань», а також усі спеціальні курси, що потребують комп'ютерного моделювання.

3. Зміст навчальної дисципліни

Розділ 1 Файлова система.

Тема 1.1. Файли і потоки. Організація потокового уведення-виведення.

Тема 1.2 Реалізація прямого доступу в C++.

Тема 1.3. Робота з консоллю.

Розділ 2 Реалізація основних принципів ООП в C++.

Тема 2.1. Об'єктно-орієнтований підхід до програмування.

Тема 2.2. Абстрагування та інкапсуляція.

Тема 2.3. Особливості роботи з класами.

Тема 2.4. Ієрархічність і успадкування.

Тема 2.5. Множинне успадкування і віртуальні класи.

Розділ 3 Поліморфізм.

Тема 3.1. Перевантаження функцій і перевантаження операторів.

Тема 3.2. Перевантаження операторів.

Тема 3.3. Механізм віртуальних функцій.

Розділ 4 Опрацювання виняткових ситуацій.

Тема 4.1. Класичні підходи до опрацювання помилок.

Тема 4.2. Механізм опрацювання винятків.

Розділ 5 Параметричний поліморфізм.

Тема 5.1. Шаблонні класи.

Тема 5.2. Шаблонні функції.

Розділ 6 . Стандартні бібліотеки.

Тема 6.1.Бібліотека STL.

Тема 6.2.Складові бібліотеки STL.

4. Навчальні матеріали та ресурси

Базова

1. Програмування та алгоритмічні мови. Частина 2. Програмування мовою C++: вступ до об'єктно-орієнтованого програмування. [Електронний ресурс]: підруч. для здобувачів ступеня

- бакалавра/за спец. 124 Системний аналіз / І. В. Назарчук; КПІ ім. Ігоря Сікорського. – Електрон. текст. дані (1 файл). – Київ : КПІ ім. Ігоря Сікорського, 2024. – 175 с.
2. Програмування та алгоритмічні мови. Частина 2. Програмування: Конспект лекцій [Електронний ресурс]: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою «Системний аналіз і управління» спеціальності 124 Системний аналіз / КПІ ім. Ігоря Сікорського; уклад.: І.В. Назарчук. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 143 с URL: <https://ela.kpi.ua/handle/123456789/48586>
 3. Програмування та алгоритмічні мови. Програмування: Методичні вказівки до комп'ютерного практикуму для студентів напряму підготовки 124 «Системний аналіз» [Електронний ресурс]: /Уклад.: І.В. Назарчук. Київ: НТУУ «КПІ», 2017. 105 с. URL: http://mmsa.kpi.ua/sites/default/files/disciplines/Програмування_та_алгоритмічні_мови/2018_09_18_programming_methods.pdf.
 4. The C++ Programming Language. Bjarne AT&T. Stroustrup. Electronic Library of Ukraine: URL: http://www.staroceans.org/e-book/The_C__Programming_Language__Stroustrup_.pdf
 5. Grady Booch. Object oriented design with applications. Benjamin/Cummings Pub. Co., Redwood City, Calif., 1990. 580 pages. URL: <https://zjnu2017.github.io/OOAD/reading/Object.Oriented.Analysis.and.Design.with.Applications.3rd.Edition.by.Booch.pdf>

Допоміжна

6. Stephen Prata. C++ Primer Plus.(Developer's Library) 6th Edition. 1440p. URL: <https://zhjwpku.com/assets/pdf/books/C++.Primer.Plus.6th.Edition.Oct.2011.pdf>
7. Войтенко В. В., Морозов А. В. С та C++. Теорія та практика. Житомир: Видавництво ЖДТУ, 2004. 324с. Бібліотека.
8. Трофименко О.Г., Прокоп Ю.В., Швайко І.Г., Буката Л.М., Косирева Л.А., Леонов Ю. Г., Ясинський В. В.. C++. Основи програмування. Теорія та практика: підручник. – Одеса, «Фенікс», 2010, 544 с. URL: <http://dspace.onua.edu.ua/handle/11300/23501>
9. Вступ до програмування мовою C++. Організація обчислень: навч. посіб. / Ю. А. Белов, Т. О. Карнаух, Ю. В. Коваль, А. Б. Ставровський. – К. : Видавничо-поліграфічний центр "Київський університет", 2012. – 175 с. URL: <http://csc.knu.ua/en/library/books/belov-24.pdf>
10. Жуковський С.С., Вакалюк Т.А. Об'єктно-орієнтоване програмування мовою C++. Навчально-методичний посібник для студентів напряму 6.040302 Інформатика.– Житомир: Вид-во ЖДУ, 2016. – 100 с. .URL: <https://core.ac.uk/download/pdf/84273987.pdf>
11. Грицюк Ю., Рак Т. Програмування мовою C++. Львів: ЛДУ БЖД, 2011. 146с. URL: https://shron1.chtyvo.org.ua/Hrytsiuk_Yurii/Obiektno-orientovane_prohramuvannia_movoju_Cpp.pdf

Інформаційні ресурси

12. Електронний кампус НТУУ «КПІ ім. І. Сікорського» [сайт] / Єдине інформаційне середовище НТУУ «КПІ ім. І. Сікорського», 2011-2020. URL: <http://campus.kpi.ua>
13. Microsoft [Електронний ресурс]//Microsoft: [сайт]/ Welcome to the Windows Console documentation. URL: <https://docs.microsoft.com/en-us/windows/console/>
14. Microsoft [Електронний ресурс]//Microsoft: [сайт]/ Microsoft C++, C, and Assembler documentation. URL: <https://docs.microsoft.com/en-us/cpp/?view=vs-2019>
15. Coursera. Дистанційний онлайн курс Coding for Everyone: C and C++ URL: <https://www.coursera.org/specializations/coding-for-everyone>
16. Coursera. Дистанційний онлайн курс Object-Oriented Data Structures in C++. URL: <https://www.coursera.org/learn/cs-fundamentals-1>
17. Coursera. Дистанційний онлайн курс Object Oriented Programming. URL: <https://www.coursera.org/specializations/object-oriented-programming-s12n>
18. ДСТУ 3008-2015. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення. [Чинний від 2015-22-07]. Вид. офіц. Київ: 2016, 31с.
19. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. [Чинний від 2016-01-07] Вид. офіц. Київ, 2016, 20с. (Інформація та документація).

5. Методика опанування навчальної дисципліни(освітнього компонента)

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу та завдання на СРС)
1	Розділ 1. Файли і потоки. Тема 1.1. Порядок роботи з файлом. Основні режими відкриття. Перевірка коректності. Закриття файлу. Робота з текстовими файлами Завдання на СРС: Динамічні структури даних [7]
2	Тема 1.2. Послідовний доступ при роботі з файлами. Реалізація прямого доступу. Редагування вмісту файлу. Робота з двійковими файлами. Завдання на СРС: Керування файлами. Функції аналізу помилок. Керування буферизацією. Знищення і перейменування файлу[2,8,9,13]
3	Тема 1.3. Робота в консолі. Текстовий режим. Дескриптор. Буфер консолі. Колір. Курсор. Робота з клавіатурою. Вікна[13]
4	Розділ 2. Реалізація основних принципів ООП в С++. Тема 2.1. Розвиток об'єктно-орієнтованої парадигми. Поняття об'єкта. Передача повідомлень об'єктам. Основні терміни. Вибір мови програмування. Складові об'єктного підходу.
5	Тема 2.2. Абстрагування та інкапсуляція. Визначення абстракції. Опис абстракції. Поведінка об'єкта. Інкапсуляція Завдання на СРС: Робота в консолі. Дескриптор вікна. Алфавітно-цифровий режим консольного вікна. [2,13]
6	Тема 2.3. Особливості роботи з класами в С++. Класифікація функцій-членів. Конструктори. Деструктори. Способи створення об'єкта. Копіювання об'єктів. Селектори. Модифікатори. Статичні члени класу
7	Тема 2.4. Ієрархічність і успадкування. Ієрархічність та її різновиди. Одиночна спадковість. Взаємозв'язок базового та похідного класів. Організація доступу. Конструктори похідних класів. Ієрархія класів. Перевизначення функцій в похідних класах. Завдання на СРС: Робота в консолі. Графічний режим консольного вікна. Організація меню [2,13]
8	Тема 2.5. Множинне успадкування і віртуальні класи. Множинне успадкування. Особливості конструювання. Можливі неоднозначності при множинному успадкуванні. Віртуальні базові класи
9	Розділ 3. Поліморфізм. Тема 3.1 . Реалізація поліморфізму в ООП. Перевантаження функцій і перевантаження операторів. Дружні класи і дружні функції. Оператор присвоєння. Оператор індексування. Завдання на СРС: Можливості стандартної бібліотеки шаблонів STL. Контейнер вектор[4,7,8]
10	Тема 3.2. Перевантаження операторів. Перевантаження унарних операторів. Перевантаження бінарних операторів. Перевантаження операторів введення – виведення. Перетворення типів.
11	Тема 3.3. Механізм віртуальних функцій. Механізм раннього зв'язування. Пізні зв'язування. Механізм пізнього зв'язування. Особливості використання віртуальних функцій. Абстрактні класи.
12	Розділ 4 Опрацювання виняткових ситуацій. Тема 4.1. Класичні підходи до опрацювання помилок. Перевірка тверджень. Винятки. Завдання на СРС: Принципи опрацювання асинхронних переривань[1,14]
13	Тема 4.2. Механізм опрацювання винятків. Ієрархії винятків. Неперехоплені винятки. Стандартні винятки. Загальні правила перехоплення винятків.
14	Розділ 5 Параметричний поліморфізм.

	Тема 5.1. Шаблонні класи. Опис шаблону класу в C++. Інстанціювання. Параметри шаблонів. Статичні члени. Шаблони і успадкування. Механізм створення виконуваного коду шаблону.
15	Тема 5.2. Шаблони функцій. Перевантаження шаблонів функцій. Особливості використання шаблонів функцій. Параметри умовчання для шаблонів. Дружні функції.
16	Розділ 6 . Стандартні бібліотеки. Тема 6.1. Бібліотека STL. Загальні принципи будови і використання основних складових бібліотеки. Порівняльна характеристика контейнерів. Ітератори.
17	Тема 6.2.Складові стандартної бібліотеки. Послідовні контейнери і робота з ними. Базові послідовні контейнери. Адаптери базових послідовних контейнерів.. Завдання на СРС: Можливості стандартної бібліотеки шаблонів STL. Алгоритми. Використання об'єктів-функцій.[3,7]
18	Тема 6.2. Складові стандартної бібліотеки. Асоціативні контейнери і робота з ними. Алгоритми. Об'єкти-функції. Порівняльна характеристика алгоритмів.

Методичні рекомендації щодо вивчення матеріалу перелічених тем надані у конспекті лекцій [1].

У другому семестрі передбачено дев'ять практичних занять, виконання дев'яти лабораторних робіт (комп'ютерних практикумів) та курсової роботи. Виконання циклу лабораторних робіт забезпечує формування практичного досвіду розробки, створення та налагодження програмного забезпечення з використанням об'єктно-орієнтованого підходу до програмування на прикладі C++ .

№ з/п	Назва теми практичної роботи	Кількість годин
1	Порядок роботи з файлами. Відміни у використанні текстових та бінарних файлів. Реалізація прямого і послідовного доступу.	2
2	Особливості використання текстового і графічного режимів консолі.	2
3	Конструювання об'єктів конкретних класів. Режими доступу.	2
4	Об'єктні і статичні члени класу. Агрегація. Копіювання.	2
5	Реалізація успадкування. Конструювання об'єктів похідних класів.	2
6	Перевантаження функцій та перевантаження операторів	2
7	Розгляд механізму віртуальних функцій.	2
8	Порядок створення шаблонних функцій. Винятки і їх застосування.	2
9	Засоби стандартної бібліотеки шаблонів.	2

Методичні рекомендації і варіанти завдань лабораторних робіт надані у навчальному посібнику [2]. Основні завдання цього циклу робіт:

№ з/п	Назва теми лабораторної роботи (комп'ютерного практикуму)	Кількість годин
1	Створення та використання текстових і бінарних файлів.	2
2	Робота в консольному вікні.	2
3	Конструювання об'єктів конкретних класів з використанням агрегації	2
4	Використання агрегації, статичних членів класу і копіювання.	2
5	Реалізація успадкування	2
6	Реалізація перевантаження операторних функцій	2
7	Використання механізму віртуальних функцій	2
8	Побудова шаблонних функцій з використанням механізму перехоплення винятків.	2
9	Використання стандартних контейнерів і алгоритмів бібліотеки STL.	2

6. Самостійна робота студента

Самостійна робота студентів передбачає виконання лабораторних робіт, оформлення звітів до кожної з них відповідно до вимог чинного законодавства України зі стандартизації, підготовка до захистів, підготовка і виконання модульної контрольної роботи (поділеної на дві частини за темами), а також самостійне опрацювання тем, що не були розглянуті або були недостатньо повно розглянуті у лекційному курсі.

Перелік тем, винесених на самостійне опрацювання у другому семестрі.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	Оформлення звітів про виконання комп'ютерних практикумів відповідно до вимог чинного законодавства України зі стандартизації[2,3,10,18,19]	18
2	Виконання тематичних контрольних робіт	4
3	Динамічні структури даних [5]	4
4	Керування файлами. Функції аналізу помилок. Керування буферизацією. Знищення і перейменування файлу [2,7,8,14]	4
5	Робота в консолі. Дескриптор вікна. Алфавітно-цифровий режим консольного вікна. [2,13]	6
6	Робота в консолі. Графічний режим консольного вікна. Організація меню[2,13]	4
7	Можливості стандартної бібліотеки шаблонів STL. Контейнери вектор, список, асоціативний масив [1,4,10,14]	2
8	Можливості стандартної бібліотеки шаблонів STL. Адаптери базових контейнерів. [1,4,9,14]	2
9	Можливості стандартної бібліотеки шаблонів STL. Використання об'єктів-функцій. [1,4,8,14]	2

Серед робіт, винесених на самостійне опрацювання, може бути рекомендоване проходження одного з рекомендованих дистанційних курсів: п.п. 15, 16, 17.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

- **Правила відвідування занять:** відвідування лекцій, практичних та лабораторних робіт (комп'ютерних практикумів) є обов'язковим. У разі відсутності студент зобов'язаний самостійно засвоїти матеріал, що викладався на пропущених заняттях (в GoogleClassroom презентація та посилання на відео в YouTube), практикуми мають бути відпрацьовані. У разі пропуску великої кількості лекційних занять (більше 30%) студент отримує додаткові питання під час поточного та семестрового контролю;
- **правила поведінки на заняттях:** активність на лекціях та практикумах заохочується додатковими балами до рейтингу;
- **правила захисту лабораторних робіт:** виконання лабораторної роботи (комп'ютерного практикуму) РКП залежно від складності може бути оцінено максимально в 10 балів, градації балів вираховують відповідно до РСО:

- «відмінно», абсолютно коректне та вчасне виконання індивідуального завдання з підготовки до РКП з виконанням вимог щодо інтерфейсу та представлення результатів, правильне та вчасне оформлення протоколу, виконання індивідуального аудиторного завдання за одне аудиторне заняття, демонстрація вільного володіння теоретичним матеріалом при показі готового результату та при експрес-тестуванні – 9-10 балів;

- «добре», коректне та вчасне виконання індивідуального завдання з підготовки до РКП з можливими незначними зауваженнями, які були виправлені безпосередньо на занятті, майже правильне та вчасне оформлення протоколу, виконання індивідуального аудиторного завдання за

одне аудиторне заняття, демонстрація володіння матеріалом при показі готового результату на рівні 75-90% – 7-8 балів;

- «задовільно», невчасне або з істотними неточностями виконання індивідуального завдання з підготовки до РКП, можливе невчасне оформлення протоколу, можливе невчасне виконання аудиторного індивідуального завдання (не за одне заняття), відповідь на половину теоретичних питань з теми даної роботи під час її демонстрації та тестування – 5-6 балів);

- Кращим студентам може додаватися 1 заохочувальний бал за розв'язування задачі підвищеної складності під час захисту РКП.

- **правила призначення заохочувальних та штрафних балів:**

- участь у модернізації робіт з комп'ютерного практикуму;
- доповіді на наукових студентських семінарах, конференціях, якщо робота мала відношення до програмування;

- участь у міжфакультетській олімпіаді з програмування, Всеукраїнських та міжнародних студентських олімпіадах з програмування любого рівня.

За їх виконання студент може отримати додатково до рейтингу максимально 10 балів.

- **політика дедлайнів та перескладань:**

- за кожний тиждень запізнення зі здачею чергової РКП оцінка може бути знижена на один бал, але не більше 2 балів (зараз цей пункт не діє);

- допустиме одне перескладання кожного контрольного заходу (захист РКП, контрольне тестування або МКР), якщо студент пропустив його з поважної причини (для визначення поважної причини документальне підтвердження не потрібне).

- **політика щодо академічної доброчесності:**

одинакові розв'язки у двох або більше студентів, несанкціоноване використання сторонніх джерел під час контрольних заходів – результати даного заходу будуть анульовані без можливості перескладання.

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

Поточний контроль:

- демонстрація роботи розробленої програми разом з експрес-опитуванням під час аудиторної роботи – до 5 балів (із 10 за КП); (всього $7*5=35$)

- захист за темою заняття (теорія) до 5 балів, (кп1.1 і кп1.2 оформлюємо як єдину роботу з двох частин і захист також як єдиної роботи – до 5 балів сумарно); захист може проходити у вигляді тесту за одну або дві роботи, оцінка відповідно до 5 або до 10 балів;(всього $6*5=30$)

- дві тематичні контрольні роботи (КР1 та КР2), кожна максимально може бути оцінена в 15 балів так:

- «відмінно», повна відповідь (не менше 90% потрібної інформації) – 13-15 балів;

- «дуже добре», «добре», достатньо повна відповідь (не менше 75% потрібної інформації), або повна відповідь з незначними зауваженнями – 11-12 балів;

- «задовільно», «достатньо», неповна відповідь (не менше 60% потрібної інформації) та незначні помилки – 7-10 балів;

- «незадовільно», незадовільна відповідь (не відповідає вимогам на 3 бали) – 0 балів;

- захист КР1 – до 5 балів – тільки у вигляді опитування. (всього за контрольні роботи $15+5+15=35$)

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Семестровий контроль: залік, може бути виставлений автоматично, якщо здані і захищені всі роботи КП і КР і сумарний бал перевищує 59. Якщо у студента всі види робіт виконані, а сумарний бал від 45 до 60, студент обов'язково має здавати залік. Якщо студент не задоволений рейтинговою оцінкою, він може анулювати результати за КР і здавати залік, на якому зможе отримати максимально 35 балів.

Умови допуску до семестрового контролю: зарахування усіх робіт КП, поточний рейтинг більше 40 балів.

Оцінювання результатів семестрового контролю побудоване на Положенні про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського (<https://osvita.kpi.ua/node/37>).

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

<i>Кількість балів</i>	<i>Оцінка</i>
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

9. Додаткова інформація з дисципліни (освітнього компонента)

- *перелік питань, які виносяться на семестровий контроль;* відповідають переліку основних питань, перелічених в темах п.5.
- *можливість зарахування сертифікатів проходження дистанційних чи онлайн курсів за відповідною тематикою:* можуть бути зараховані частково (перелік у списку інформаційних ресурсів: 13, 14, 15).

Робочу програму навчальної дисципліни (силабус):

Складено старший викладач, Назарчук Ірина Василівна

Ухвалено кафедрою ММСА (протокол № 13 від 05.06.2024)

Погоджено Методичною комісією факультету (протокол № 10 від 24.06.2024)