

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії
/назва факультету/

Кафедра інформатики і математичного моделювання
/назва кафедри /

ЗАТВЕРДЖУЮ

Декан факультету

Ігор БАРАН

(підпис)

(прізвище та ініціали)

«02» 09 2024 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ПРОГРАМУВАННЯ

/назва дисципліни/

галузь знань 12 "Інформаційні технології"
/шифр і назва галузі знань/

рівень вищої освіти перший (бакалаврський)
/назва/

спеціальність 122 "Комп'ютерні науки"
/шифр і назва/

освітня програма Комп'ютерні науки
/назва/

спеціалізація Комп'ютерні науки
/назва/

вид дисципліни обов'язкова дисципліна циклу професійної підготовки
/обов'язкова/вибіркова/

Тернопіль – 2024 рік

Робоча програма з навчальної дисципліни "Програмування"

(назва дисципліни)

для студентів факультету комп'ютерно-інформаційних систем
і програмної інженерії

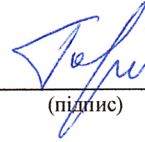
(назва факультету)

Розробники:

доцент кафедри інформатики і математичного
моделювання ,

кандидат технічних наук, доцент

(посада, науковий ступінь та вчене звання)



(підпис)

/ Юрій ГЛАДЬО /

(ініціали та прізвище)

Робоча програма розглянута та схвалена на засіданні
кафедри інформатики і математичного моделювання

(назва)

Протокол від «26» серпня 2024 року № 1

Завідувач кафедри, к.ф.-м.н.

(підпис)



/ Михайло МИХАЙЛИШИН /

(ініціали та прізвище)

Робоча програма розглянута та схвалена НМК
факультету комп'ютерно-інформаційних систем і програмної інженерії

(назва)

Протокол від «03» вересня 2024 року № 1

Секретар НМК, к.т.н.



(підпис)

/ Богдана МЛИНКО /

(ініціали та прізвище)

Робоча програма погоджена:

Спеціальність 122 "Комп'ютерні науки"

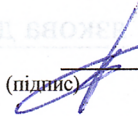
(шифр і назва)

освітня програма Комп'ютерні науки

(назва)

Завідувач випускової кафедри, к.т.н.

(підпис)

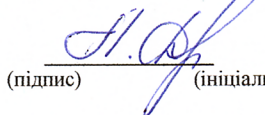


/ Ігор БОДНАРЧУК /

(ініціали та прізвище)

Гарант освітньої програми, к.т.н.

(підпис)



/ Леся ДМИТРОЦА /

(ініціали та прізвище)

1. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Показник	Всього годин	
	Денна форма навчання	Заочна (дистанційна) форма навчання
Кількість кредитів/годин	11/330	11/330
Аудиторні заняття, год.	118	24
Самостійна робота, год.	212	306
Аудиторні заняття		
- лекції, год.	50	12
- лабораторні, год.	68	12
- практичні заняття, год.	-	-
- семінарські, год.	-	-
Самостійна робота		
- підготовка до лабораторних занять	64	90
- опрацювання теоретичного матеріалу, що виклаався на лекції	64	90
- опрацювання окремих розділів програми, які не виносяться на лекції	60	90
- виконання контрольних завдань	-	-
- виконання індивідуальних завдань	-	-
- виконання курсових проектів (робіт)	-	-
- підготовка та складання заліків, екзаменів, контрольних робіт, тестування	24	36
Вид підсумкового контролю - екзамен - 1 семестр, екзамен - 2 семестр		

Частка годин самостійної роботи студента:

- денна форма навчання - 64,2 %;
- заочна (дистанційна) форма навчання - 91,7%.

2. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

2.1. **Мета** дисципліни «Програмування» полягає у вивченні методів та способів підготовки і розв'язування задач на сучасних ПК.

2.2. **Завдання** дисципліни – набуття студентами теоретичних знань і практичних навичок у побудові обчислювальних алгоритмів, та створенню на їх основі програм для розв'язання прикладних задач.

2.3. У результаті вивчення курсу студент повинен

знати:

- сучасні технічні засоби взаємодії з ПК;
- основи алгоритмізації прикладних задач;
- методи розробки алгоритмів і складання програм на сучасних мовах програмування;
- методи відладки і розв'язування задач на ПК в різних режимах і операційних системах.

вміти:

- працювати з сучасними технічними засобами взаємодії з ПК;
- проектувати компоненти програмного забезпечення;
- будувати алгоритми різних типів;
- створювати програми на сучасних мовах програмування на основі раніше розроблених алгоритмів;
- здійснювати відладку та тестування програм;
- проводити тестування розроблених програм;
- проектувати людино-машинний інтерфейс інформаційних систем.

2.3. Вивчення навчальної дисципліни передбачає формування та розвиток у студентів наступних компетентностей:

-загальних:

- ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.
- ЗК2. Здатність застосовувати знання у практичних ситуаціях.
- ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.
- ЗК6. Здатність вчитися й оволодівати сучасними знаннями.
- ЗК11. Здатність приймати обґрунтовані рішення.

-спеціальних (фахових, предметних):

СК3. 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.

СК8. 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.

За результатами вивчення дисципліни студент повинен продемонструвати такі програмні результати навчання

ПР5. 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.

ПР9. 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.

3. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

3.1. ЛЕКЦІЙНІ ЗАНЯТТЯ

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
1	Тема 1. Поняття алгоритму , зображення алгоритму, застосування блоків на блок-схемах. Алгоритми найпростіших обчислювальних процесів.	2	1
2	Тема 2. Алгоритм табулювання функції. Лінійний обчислювальний процес. Розгалужений обчислювальний процес. Алгоритми найпростіших обчислювальних процесів. Циклічний обчислювальний процес. Обчислення суми і добутку.	2	
3	Тема 3. Мова структурного програмування Сі. Алфавіт та основні конструкції мови. Загальна структура програми. Оголошення змінних і констант. Вирази числового типу. Основні арифметичні операції.	2	1
4	Тема 4. Вирази послідовного обчислення та умовного обчислення. Операція присвоєння. Операції інкременту та декременту. Приклад простої програми.	2	1
5	Тема 5. Оператори мови Сі. Складений та порожній оператори. Оператор обчислення виразу. Оператор безумовного переходу goto . Умовний оператор if (оператор розгалуження). Операції відношення та логічні операції. Оператор вибору switch . Програмування алгоритмів розгалуженої структури	2	1
6	Тема 6. Реалізація алгоритмів циклічної структури. Оператори циклу типу перерахунку for , з передумовою while , з післяумовою do..while . Оператори break і continue . Приклад програми табулювання функції	2	1
7	Тема 7. Оголошення індексних змінних на мові Сі. Різні види задання масивів. Порядок використання індексних змінних в програмах. Організація вводу-виводу індексних змінних. Приклад програми обробки масивів.	2	1
8	Тема 8. Сортування масивів, опрацювання матриць, пошук максимуму, накопичення суми, кількості та добутку елементів за певною умовою.	2	
9	Тема 9. Поняття вказівника та адреси даних. Створення динамічних змінних та порядок їх використання у програмах. Зв'язок вказівників та масивів. Динамічне виділення пам'яті, операції new , delete , malloc , free . Типові помилки при роботі із вказівниками. Приклади програм.	4	1

10	Тема 10. Підпрограми на мові Сі. Оголошення та структура функцій. Локальні та глобальні змінні. Параметри-значення і параметри-змінні. Оператор завершення функції return . Передача параметрів вказівникового типу до функцій. Класи пам'яті.	2	1
11	Тема 11. Оператор звернення до функції. Взаємодія фактичних та формальних параметрів. Приклади запису функцій. Поняття про прототипи функцій. Реалізація рекурсивного виклику функції.	2	
12	Тема 12. Директиви препроцесора #include , #define , #undef , #if , #elif , #else , #endif , #ifdef , #ifndef , #line . Поняття модульності програми. Включення файлів модулів (бібліотек) у програму. Проблема подвійного включення величин. Умовна компіляція.	4	
13	Тема 13. Символьні та стрічкові величини на мові Сі. Основні стандартні функції для роботи з символьними та стрічковими величинами. Приклади програм опрацювання текстових величин.	2	1
14	Тема 14. Складні структури даних struct та union . Бітові поля та величини перерахованого типу. Застосування елементів складних структур у програмі.	2	1
15	Тема 15. Робота з файлами на мові Сі. Відкриття та закриття файла, операції вводу та виводу у файл. Стандартні функції опрацювання файлів.	2	1
16	Тема 16. Текстові та двійкові (бінарні) файли. Приклади програм опрацювання бінарних та текстових файлів. Програмування опрацювання файлу бази даних, створеного на основі файлів структур на прикладі "Телефонного довідника".	2	
17	Тема 17. Позиційні системи числення. Двійкова, вісімкова та шістнадцяткові системи числення. Їх конвертація у десяткову систему. Основні арифметичні операції над числами в різних системах числення. Запис від'ємних чисел у доповнюючому коді.	2	
18	Тема 18. Поняття про зв'язані списки. Стеки, черги, кільцеві буфери, бінарні дерева. Програмні методи представлення інформації у зв'язаних списках. Приклади типових алгоритмів і програм опрацювання зв'язаних списків.	4	
19	Тема 19. Основні поняття об'єктно-орієнтованого програмування, принципи та парадигма ООП. Засоби мови Сі++ для створення об'єктно-орієнтованих програм. Поняття про класи і їх структуру, поля, властивості та методи класу, області видимості. Конструктори та деструктори класу.	4	1

20	Тема 20. Поняття інкапсуляції, спадкування та поліморфізму. Створення класів-нащадків. Дружні класи і функції. Поліморфізм функцій класів нащадків. Поняття та реалізація абстрактних класів. Перевантаження операцій. Приклади програм.	4	
Усього годин:		50	12

3.2. ЛАБОРАТОРНІ ЗАНЯТТЯ

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
1	Вступне заняття, знайомство з завданнями дисципліни, інструктаж з техніки безпеки.	2	
2	Розробка схем алгоритмів лінійної, розгалуженої і циклічної структури.	4	2
3	Знайомство з інтегрованим середовищем розробки. Порядок реалізації Сі-програм в діалоговому режимі на персональних ПК.	2	
4	Створення програми на мові Сі опрацювання задач з лінійним обчислювальним процесом.	2	1
5	Створення програми на мові Сі опрацювання задач з розгалуженим обчислювальним процесом, реалізованим оператором умови.	4	1
6	Створення програми на мові Сі опрацювання задач з розгалуженим обчислювальним процесом, реалізованим оператором вибору.	2	
7	Створення програми на мові Сі опрацювання задач з циклічним обчислювальним процесом з визначеною кількістю повторень.	2	1
8	Створення програми на мові Сі опрацювання задач з циклічним обчислювальним процесом, який визначається умовним циклом.	2	
9	Створення програми на мові Сі опрацювання задач обробки одновимірних масивів.	2	1
10	Створення програми на мові Сі опрацювання задач обробки двовимірних масивів.	2	
11	Створення програми на мові Сі опрацювання задач опрацювання символьних величин.	2	
12	Створення програми на мові Сі опрацювання задач опрацювання текстових величин.	2	1
13	Створення програми на мові Сі опрацювання задач з використанням підпрограм-функцій.	2	1
14	Створення програми на мові Сі опрацювання задач з використанням void підпрограм.	4	

15	Створення програми на мові Сі опрацювання задач з використанням вказівників .	4	
16	Створення програми на мові Сі опрацювання задач по обробці величин комбінованого типу (структур).	4	1
17	Застосування текстових файлів для вводу та виводу результатів роботи програми	2	1
18	Застосування двійкових файлів для вводу та виводу інформації структурованого типу	2	
19	Запис чисел у двійковій та шістнадцятковій системах числення. Арифметичні операції над ними.	2	
20	Сторення програми із застосуванням рекурсивного звернення до функції	2	
21	Застосування динамічних величин різного типу у програмі на мові Сі.	4	
22	Створення програми модульного типу на мові Сі.	2	
23	Створення бібліотеки користувача на мові Сі.	2	1
24	Застосування засобів умовної компіляції для створення складної програми на мові Сі.	2	
25	Створення комплексної програми на мові Сі для опрацювання бази даних власної структури	8	1
Усього годин:		68	12

3.3. САМОСТІЙНА РОБОТА

№	Найменування робіт	Кількість годин	
		ДФН	ЗФН
1	Опрацювання лекційного матеріалу	64	90
2	Підготовка до лабораторних занять	64	90
3	Опрацювання окремих розділів програми, які не виносяться на лекції: - алгоритми опрацювання багатовимірних масивів - застосування порозрядних арифметично-логічних операцій - вивчення роботи інтегрованих середовищ для створення програмного забезпечення на мові Сі++ - вивчення функцій опрацювання стрічкових величин модуля <i>strings.h</i> - знайомство з директивами передпроцесора - поглиблене вивчення роботи з двійковими файлами в стилі потоків із застосуванням бібліотеки <i>fstream.h</i> - застосування параметрів функцій у вигляді вказівників та посилань - поглиблене знайомство з основами ООП	60	90

4	Підготовка та складання заліку, екзамену, тестування		
	- модуль №1	2	4
	- модуль №2	2	4
	- модуль №3	2	4
	- модуль №4	2	4
	- екзамен 1 семестр	8	10
	- екзамен 2 семестр	8	10
Усього годин:		212	306

4. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ СТУДЕНТІВ

1 семестр - форма підсумкового семестрового контролю – **екзамен**

Модуль 1		Модуль 2		Підсумковий контроль екзамен	Разом з дисципліни
Аудиторна та самостійна робота		Аудиторна та самостійна робота			
Теоретичний курс (тестування)	Лабораторні роботи	Теоретичний курс (тестування)	Лабораторні роботи		
20	15	20	20	25	100

2 семестр - форма підсумкового семестрового контролю – **екзамен**

Модуль 3		Модуль 4		Підсумковий контроль екзамен	Разом з дисципліни
Аудиторна та самостійна робота		Аудиторна та самостійна робота			
Теоретичний курс (тестування)	Лабораторні роботи	Теоретичний курс (тестування)	Лабораторні роботи		
20	15	20	20	25	100

5. НАВЧАЛЬНО-МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

1. Гладь Ю.Б., Хоміцький Б.В. Методичні вказівки до виконання лабораторних робіт з курсу "Програмування". Частина 1. –ТНТУ, 2022, 36 с.
2. Гладь Ю.Б., Хоміцький Б.В. Методичні вказівки до виконання лабораторних робіт з курсу "Програмування". Частина 2. –ТНТУ, 2022, 32 с.
3. Електронний навчальний курс "Програмування" (ID 1210).
4. Конспект лекцій (електронний курс).

6. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Базова

1. Трофименко О. Г. С++. Алгоритмізація та програмування : підручник / О. Г. Трофименко, Ю. В. Прокоп, Н. І. Логінова, О. В. Задерейко. 2-ге вид. перероб. і доповн. – Одеса : Фенікс, 2019. – 477 с.
2. Васильєв О. Програмування на С++ в прикладах і задачах : Навч. посіб. / – Київ : Видавництво Ліра-К, 2017. – 382 с.
3. Злобін Г.Г. Основи алгоритмізації та програмування мовою Сі : Підручник - Київ : Видавництво Каравелла , 2022, - 170 с.
4. Вступ до програмування мовою С++. Організація даних / Т. О. Карнаух, Ю. В. Коваль, М. В. Потієнко, А. Б. Ставровський. – К.: ВПЦ "Київський університет", 2015.
5. Пекарський Б.Г. Основи програмування: Навчальний посібник. Кондор, 2018. 364 с.

Додаткова

1. Белов Ю.А. Вступ до програмування мовою С++. / Ю.А. Белов, Т.О. Карнаух, Ю.В. Коваль, А.Б. Ставровський. – К.: Видавничо-поліграфічний центр «Київський університет», 2012. – 175 с.
2. Шпак З.Я Програмування мовою С. – Львів: Оріяна-Нова, 2006.-432с.
3. В.Ю.Вінник. Алгоритмічні мови та основи програмування: мова Сі. - Житомир: ЖДТУ, 2007.- 328 с.
4. Я. Глинський і ін. С++ Builder Навчальний посібник. – Львів: Деол, СПД Глинський, 2003 – 192 с.

7. ІНФОРМАЦІЙНІ РЕСУРСИ

- 1 [Prometheus: CS50 Основи програмування CS50](#)
- 2 [C++ Language](#)
- 3 [C++ reference](#)
- 4 [C++Builder](#)
- 5 [C Programming and C++ Programming](#)

8. ЗМІНИ ТА ДОПОВНЕННЯ ДО РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

[illegible]