

Чернівецький національний університет імені Юрія Федьковича

**факультет математики та інформатики
Прикладної математики та інформаційних технологій**

**СИЛАБУС
навчальної дисципліни**

Об'єктно-орієнтоване програмування

обов'язкова

Освітньо-професійні програми:

1. «Інформаційні технології та управління проектами».
2. «Технології програмування та комп'ютерне моделювання».
3. «Системний аналіз»

Спеціальності:

1. 122 «Комп'ютерні науки та інформаційні технології».
2. 113 «Прикладна математика».
3. 124 «Системний аналіз»

Галузі знань:

1. 12 «Інформаційні технології».
2. 11 «Математика та статистика».
3. 12 «Інформаційні технології»

Рівень вищої освіти перший (бакалаврський)

Факультет математики та інформатики

Мова навчання українська

Розробник:

Сопронюк Т. М., доцент кафедри прикладної математики та інформаційних технологій, кандидат фізико-математичних наук

Профайл викладача <https://amit.chnu.edu.ua/pro-kafedru/personalii/soproniuk-tetiana-mykolaivna/>

E-mail: t.sopronyuk@chnu.edu.ua

Сторінка курсу в Moodle <https://moodle.chnu.edu.ua/course/view.php?id=1901>

Онлайн-консультації: Середа з 18.00 до 19.00.

1. Анотація дисципліни (призначення навчальної дисципліни).

Курс присвячено вивченню сучасних технологій програмування на С++ (парадигми процедурного, модульного і об'єктно-орієнтованого програмування) з використанням класів, успадкування і поліморфізму. У курсі демонструється застосування об'єктно-орієнтованого підходу для розробки програмного забезпечення.

2. Мета навчальної дисципліни: поглиблене оволодіння сучасними технологіями програмування на С++ з використанням класів, успадкування і поліморфізму. Для досягнення мети студенти повинні оволодіти програмним матеріалом, написати програми, виконати модульні контрольні роботи, здати екзамен або залік.

3. Пререквізити. Для ефективності засвоєння курсу здобувач вищої освіти має вивчити дисципліну «Програмування».

4. Результати навчання.

У результаті вивчення навчальної дисципліни студент повинен

знати: поняття базового і породженого класів, ієрархії класів; методику програмування, що концентрує основну увагу на зв'язках між об'єктами, а не на деталях їхньої реалізації (інкапсуляція дає можливість об'єднання даних і алгоритмів їх обробки, в результаті чого і дані, і процедури багато в чому втрачають самостійне значення);

вміти: застосовувати теоретичні знання для створення власних класів, переважувати оператори для структур і класів, перевизначати функції, застосовувати шаблони функцій і класів, успадковувати класи, побудувати ієрархію класів, використовувати віртуальні функції і абстрактні класи. При написанні лабораторних робіт і, взагалі, програмних систем студент повинен вміти застосовувати об'єктно-орієнтований підхід.

Під час вивчення дисципліни, відповідно до освітньо-професійної програми, формуються наступні

загальні компетентності:

- ЗК01. Здатність учитися і оволодівати сучасними знаннями.
- ЗК02. Здатність застосовувати знання у практичних ситуаціях.
- ЗК03. Здатність генерувати нові ідеї (креативність).
- ЗК06. Здатність до абстрактного мислення, аналізу та синтезу.
- ЗК10. Навички у використанні інформаційних і комунікаційних технологій.

фахові компетентності:

- ФК04. Здатність розробляти алгоритми та структури даних, програмні засоби та програмну документацію.
- ФК05. Здатність проектувати бази даних, інформаційні системи та ресурси.
- ФК06. Здатність розв'язувати професійні задачі за допомогою комп'ютерної техніки, комп'ютерних мереж та Інтернету, в середовищі сучасних операційних систем, з використанням стандартних офісних додатків.
- ФК07. Здатність експлуатувати та обслуговувати програмне забезпечення автоматизованих та інформаційних систем різного призначення.
- ФК08. Здатність використовувати сучасні технології програмування та тестування програмного забезпечення.
- ФК09. Здатність до проведення математичного і комп'ютерного моделювання, аналізу та обробки даних, обчислювального експерименту, розв'язання формалізованих задач за допомогою спеціалізованих програмних засобів.
- ФК17. Здатність до використання новітніх інформаційно-комунікаційних технологій.

та отримуються наступні програмні результати навчання:

ПРН11. Вміти застосовувати сучасні технології програмування та розроблення програмного забезпечення, програмної реалізації чисельних і символьних алгоритмів.

ПРН14. Виявляти здатність до самонавчання та продовження професійного розвитку.

5. Опис навчальної дисципліни

5.1. Загальна інформація

Назва навчальної дисципліни _____												
Форма навчання	Рік	Семестр	Кількість			Кількість годин						Вид підсумкового контролю
			кредити	годин	змістови	лекції	практич	семінарс	лаборато	самостій на	індивідуа	
Денна	2	3	5	150	3	45	-	-	30	75	-	Екзамен (Прикладна математика) Залік(Інформаційні технології та управління проектами, Системний аналіз)

5.2. Дидактична карта навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
		лек	п	лаб	інд	с.р.
1	2	3	4	5	6	7
Змістовий модуль 1. Нові можливості C++, які не пов'язані з ООП						
Тема 1 Парадигма програмування. Процедурне, модульне і об'єктно орієнтоване програмування Вступ. Парадигма програмування. Процедурне програмування. Модульне програмування. Об'єктно-орієнтоване програмування. Загальна характеристика мови C++ як мови, що найбільш втілює ідеї об'єктно-орієнтованого програмування.	19	5		4		10
Тема 2. Розширені можливості мови C++, що не пов'язані з класами Прототипи функцій, аргументи по замовчуванню, посилання. Модифікатори const і volatile. Функціональний запис перетворення типів. Доступ до глобальних змінних, які закриті локальними. Функції, що підставляються. Оператори динамічного	31	10		6		15

розподілу пам'яті. Перезавантаження функцій, шаблони функцій. Перезавантаження операторів для структур. Перезавантаження операцій потокового вводу і виводу для структур. Приклади перезавантаження операцій і функцій, використання шаблонів.					
Разом за змістовним модулем 1	50	15		10	25
Змістовий модуль 2. Інкапсуляція. Поняття класу					
Тема 3. Класи і об'єкти. Створення і знищення об'єктів Поняття класу та його визначення. Специфікатори public, private. Доступ до відкритих членів класу. Функції доступу до захищених членів класу. Створення об'єктів. Статичні члени класу. Дружні і складові функції. Дружні класи. Ініціалізація та знищення об'єктів. Автоматичне і динамічне виділення пам'яті під об'єкти класу. Конструктори і деструктори.	22	8		4	10
Тема 4. Операції над класовими об'єктами Перевизначення операцій для класів. Бінарні і унарні операції. Пріоритети при перевизначенні операцій. Присвоєння і ініціалізації. Перезавантаження операторів виклику функції, індексування. Стандартні об'єкти потокового вводу-виводу C++. Перезавантаження операцій "<<", ">>" для класу користувача. Приклади проектування і побудови класів.	28	7		6	15
Разом за змістовним модулем 2	50	15		10	25
Змістовий модуль 3. Успадкування і поліморфізм					
Тема 5. Успадкування. Ієрархія класів. Поняття похідного та базового класів. Захищені члени класу. Співвідношення атрибутів доступу в базовому і похідному класах. Ієрархія класів. Віртуальні базові класи. Контейнеризація Конструктори і деструктори похідних класів. Перетворення вказівників на об'єкти класів. Приклади успадкування.	14	5		4	9
Тема 6. Поліморфізм. Віртуальні функції. Віртуальні функції, їх відмінність від звичайних функцій-членів. Віртуальні деструктори. Абстрактні функції і абстрактні класи. Поліморфні функції. Багаточисельне успадкування. Створення бібліотеки класів графічних фігур.	18	5		3	8
Тема 7. Бібліотеки класів і шаблонів C++ Простори імен. Огляд можливостей бібліотек STL, BOOST, MTL. Робота з рядками засобами стандартної бібліотеки та без них. Узагальнене програмування і стандартна бібліотека шаблонів STL. Приклади використання бібліотеки STL	18	6		3	8
Разом за змістовним модулем 3	50	15		10	25

Усього годин	150	45		30		75
--------------	-----	----	--	----	--	----

5.3. Зміст завдань для самостійної роботи

№ п/п	Назва теми	Кількість годин /форми контролю
1	Виняткові ситуації	Застосування в лабораторних роботах
2	Робота з потоками	Застосування в лабораторних роботах
3	Графічні можливості	Застосування в лабораторних роботах
4	Створення віконних інтерфейсів (GUI)	Застосування в лабораторних роботах

5.4. Теми лабораторних занять

№ модуля	Зміст модуля	Кількість годин	Кількість балів
1	· Лабораторна робота 1. Структури. Зовнішні функції і функції-члени структури 4год.(3+1)	10	6(3+3)
	· Лабораторна робота 2. Перезавантаження операцій і функцій для структур. Використання шаблонів (2год.)		5
	· Лабораторна робота 3. Структури і класи 4год.(3+1).		8 (4+4)
	· Тести		10
2	· Лабораторна робота 4. Створення класів. Конструктори, деструктори, дружні функції і члени класу. Перезавантаження операцій для класів (6год.)	12	15(5+5+5)
	· Створення умови задачі для лабораторної роботи 5 (успадкування або агрегація (0.5год.))		2
	· Лабораторна робота 5. Проектування ієрархії класів. Успадкування. Агрегація (5год.)		11
	• Контрольна робота/задача. Створення класів (0.5год.)		10
3	· Лабораторна робота 6. Абстрактні класи (4год.)	8	11
	· Лабораторна робота 7. Створення графічного інтерфейсу до лабораторної роботи 4 , 5 або 6 (4год.)		7
	· Модуль контроль (тести)		15

6. Система контролю та оцінювання Види та форми контролю

Під час проведення лекцій використовуються пасивний та активний методи навчання. Консультаційна робота. Під час виконання студентами лабораторних робіт використовується активні методи навчання. Проведення модульних контрольних робіт та навчальна робота під час прийому лабораторних робіт.

Методи контролю

1. Тестові завдання в 1-му модулі засобами системи Moodle.
2. Контрольна робота в 2-му модулі.
3. Оцінювання на лабораторних заняттях в 3-х модулях.
4. Тестові завдання на екзамену/заліку засобами системи Moodle.

Захист та критерії оцінювання лабораторної роботи

- Здача лабораторної роботи проводиться під час заняття згідно з календарним планом.
- Для захисту лабораторної роботи кожен студент має самостійно виконати лабораторну роботу і здати її викладачу **на занятті**.
- Не допускається заочне прийняття програм (електронною поштою) без запуску програм з різними вхідними даними.
- Під час здачі програми викладач зобов'язаний перевіряти здатність студента орієнтуватися у власній програмі, пропонуючи йому виконати нескладні зміни, розраховані на 5-10 хвилин поточного заняття.
- При необхідності виконання частини завдання або усього завдання у **робочому** зошиті, бали виставляти у зошиті, вказуючи число і підпис.
- Під час здачі лабораторної роботи студент повинен
 - вміти пояснити постановку задач, які розв'язувались в лабораторній роботі; алгоритм розв'язування задач; програмну реалізацію завдання;
 - продемонструвати розуміння програми та обґрунтувати зроблені висновки;
 - відповісти на питання, які належать до виконання лабораторної роботи та додаткові теоретичні питання, якщо розданий перелік таких питань.
- Якщо студент не розуміє алгоритму розв'язання задачі, не орієнтується в програмній реалізації, але є у наявності правильно виконувана програма, то робота зараховується не більше як на 30%.
- Якщо студент розуміє задачу і алгоритм її виконання, але не орієнтується (слабо орієнтується) в практичній частині (програмній реалізації), то оцінка знижується до 50%.
- Якщо програма не працює, або працює частково і студент може пояснити алгоритм, роботу оцінювати частково, в залежності від об'єму і якості коду.
- Кількість балів за лабораторну роботу визначає викладач в процесі здачі. Оцінка повідомляється студенту.
- За невчасний захист лабораторних робіт у межах модуля допускається знімати по одному балу за кожне прострочене заняття, якщо робота оцінюється до 10 балів, і по 1,5-2 бали, якщо робота оцінюється в межах від 11 до 20 балів, але не більше половини балів.

Екзамен проводиться проводиться засобами дистанційної системи Moodle.

Складові частини оцінки:

1) тести на 25 балів:

- 10 балів (тематика першого модуля);

- 15 балів (тематика другого і третього модулів).
- 2) програмно згенерована випадкова контрольна задача на тему “Створення класів” на 10 балів.

У сумі з лабораторними роботами (65 балів) це загалом складатиме 100 балів.

Розподіл балів, які отримують студенти
для заліку (системний аналіз, комп'ютерні науки)

Поточне тестування та самостійна робота					Залік	Сума
Змістовий модуль №1		Змістовий модуль № 2		Змістовий модуль № 3		
T1	T2	T3	T4	T5	T6	T7
6	5	8	15	13	11	7

T1, T2 ... T7 – теми змістових модулів.

У таблиці зазначені бали за лабораторні роботи. Крім того, оцінюються тести у 1-му модулі (10 балів) і контрольна робота у 2-му модулі (10 балів).

для екзамену (прикладна математика)

Поточне тестування та самостійна робота					Підсумковий тест (екзамен)	Сума
Змістовий модуль №1		Змістовий модуль № 2		Змістовий модуль № 3		
T1	T2	T3	T4	T5	T6	T7
6	5	8	15	13	11	7

T1, T2 ... T7 – теми змістових модулів.

Критерії оцінювання результатів навчання з навчальної дисципліни

Сума балів за всі види навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену	для заліку
90 – 100	A	відмінно	зараховано
80-89	B	добре	
70-79	C		
60-69	D	задовільно	
50-59	E		
35-49	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання

0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни
------	---	--	---

7. Рекомендована література

1. Сопронюк Т.М. Об'єктно-орієнтоване програмування на C++ : навч. посібник / Т. М. Сопронюк. – Чернівці : Чернівецький національний ун-т, 2013. – 175 с. (з грифом МОНУ, лист №1/11-17600 від 18.11.2013)
2. Sopronyuk T. Object-oriented programming in C++: Textbook / Translated by Nonna Shulga: CreateSpace, 2014. – 130 p. ISBN-13: 978-1502520906 (переклад навчального посібника з грифом МОНУ: Сопронюк Т.М. Об'єктно-орієнтоване програмування на C++ : навч. посібник / Т. М. Сопронюк. – Чернівці : Чернівецький національний ун-т, 2014. – 176 с.)
3. 150 C++ Programming Assignments. Variants of tasks & Examples of the code: Textbook / Authored by Tatyana Sopronyuk, Translated by Nonna Shulga: CreateSpace, 2015. – 73 p. ISBN-13: 978-1515254065 (CreateSpace-Assigned) ISBN-10: 1515254062.
4. Глинський Я.М., Анохін В.Є., Рязьська В.А. C++ і C++ Builder. -Львів: Деол, 2003.- 192 с.
5. Ткачук В.М. Програмування на C++: Лабораторний практикум. Видавництво Прикарпатського національного університету імені Василя Стефаника, 2011.- 160с
6. ГрицюкЮ.І., Рак Т.Є. Програмування мовою C++ : навчальний посібник. - Львів : Вид-во Львівського ДУ БЖД, 2011. - 292 с.
7. Ю. А. Белов, Т. О. Карнаух, Ю. В. Коваль, А. Б. Ставровський. Вступ до програмування мовою C++. Організація обчислень : навч. Посіб - К.: Видавничополіграфічний центр "Київський університет 2012. - 175 с.
8. [C++ common knowledge : essential intermediate programming/ C++ \(Computer program language\)](#), Dewhurst, Stephen C. Addison-Wesley, Upper Saddle River, N. J.: 2005.
9. [C++ programming cookbook Herb Schildt's C++ programming cookbook / C++ \(Computer program language\)](#), Schildt, Herbert. McGraw-Hill, New York: c2008.
10. [Problem solving with C++: The object of programming/ C++ \(Computer program language\)](#). Savitch, Walter. Pearson Addison Wesley, Boston: 2005. Fifth Edition (International ed.)
11. [C++ programming : From Problem Analysis to Program Design / C plus plus programming.](#): Malik, D S. Course Technology, Boston, MA : c2009. Fourth Edition.
12. [Problem solving with C++/](#) Savitch, Walter J, 1943- Pearson/Addison-Wesley, Boston : c2006.Sixth Edition.

8. Інформаційні ресурси

1. Google C++ Style Guide [Електронний ресурс]. – Режим доступу: <https://google.github.io/styleguide/cppguide.html>
2. The Stanford University C++ Style Guide [Електронний ресурс]. – Режим доступу: <https://hownot2code.com/2017/01/18/the-stanford-university-c-style-guide/>
3. 7 Best C++ Books For Beginners to Advanced Programmers Guide [Електронний ресурс]. – Режим доступу: <https://levelup.gitconnected.com/7-best-c-books-for-beginners-to-advanced-programmers-c7356471cad8>

Додаток

Методичне забезпечення

- **Сопронюк Т.М.** Об'єктно-орієнтоване програмування на C++ : навч. посібник / Т. М. Сопронюк. – Чернівці : Чернівецький національний ун-т, 2013. – 175 с. (з грифом МОНУ, лист №1/11-17600 від 18.11.2013)

- **Sopronyuk T.** Object-oriented programming in C++: Textbook / Translated by Nonna Shulga: CreateSpace, 2014. – 130 p. ISBN-13: 978-1502520906 (переклад навчального посібника з грифом **МОНУ**: Сопронюк Т.М. Об'єктно-орієнтоване програмування на C++ : навч. посібник / Т. М. Сопронюк. – Чернівці : Чернівецький національний ун-т, 2014. – 176 с.
- **Sopronyuk T.** 150 C++ Programming Assignments. Variants of tasks & Examples of the code: Textbook / Authored by Tatyana Sopronyuk, Translated by Nonna Shulga: CreateSpace, 2015. – 73 p. ISBN-13: 978-1515254065 (CreateSpace-Assigned) ISBN-10: 1515254062.
- **Sopronyuk T.** C++ Programming: Theory and Assignments // Authored by Tatyana Sopronyuk, Translated by Nonna Shulga: Independently published, 2020. – 177 p. ISBN-13: 979-8651045624,
- Тестові завдання (Система Moodle)
- Презентації лекцій (Система Moodle)
- Відео-лекції на Google диску

