

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
«ПРИКАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАСИЛЯ СТЕФАНІКА»



Факультет/інститут **фізико-технічний**

Кафедра **комп'ютерної інженерії і електроніки**

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ПРОГРАМУВАННЯ

Освітня програма **Комп'ютерна інженерія, Прикладна фізика та наноматеріали, Фізика та астрономія та інші**

Спеціалізація (за наявності) _____

Спеціальність **123 Комп'ютерна інженерія**

105 Прикладна фізика та наноматеріали

104 Фізика та астрономія

та інші

Галузь знань **12 Інформаційні технології**

10 Природничі науки та інші

Затверджено на засіданні кафедри
Протокол № __ від “_” ____ 2022 р.

м. Івано-Франківськ – 2022 р.

Зміст

1. Загальна інформація.....	3
2. Опис дисципліни.....	3
3. Структура курсу.....	5
4. Система оцінювання курсу.....	8
5. Оцінювання відповідно до графіку навчального процесу.....	8
6. Ресурсне забезпечення.....	8
7. Контактна інформація.....	9
8. Політика навчальної дисципліни.....	10

1. Загальна інформація

Назва дисципліни	Програмування
Освітня(і) програма(и)	Комп'ютерна інженерія Прикладна фізика та наноматеріали Фізика та астрономія та інші
Спеціалізація (за наявності)	
Спеціальність	123 Комп'ютерна інженерія 105 Прикладна фізика та наноматеріали 104 Фізика та астрономія та інші
Галузь знань	10 Природничі науки 12 Інформаційні системи та інші
Освітній рівень	бакалавр
Статус дисципліни	Нормативна, вибіркова
Курс / семестр	1 курс, 1 семестр (для нормативної) будь-який курс та семестр при виборі здобувача освіти
Розподіл за видами занять та годинами навчання (якщо передбачені інші види, додати)	9 кредитів Лекції – 44 год. Лабораторні заняття – 46 год. Самостійна робота – 180 год.
Мова викладання	Українська
Посилання на сайт дистанційного навчання	https://d-learn.pnu.edu.ua

2. Опис дисципліни

Мета та цілі курсу:	<p>Вивчення студентами основ програмування на мові високого рівня C++, методів проектування та створення програм згідно сучасних технологій програмування, теоретичних і практичних методів побудови алгоритмів для розв'язування прикладних та спеціалізованих задач на мові програмування C++, та створення готових програмних продуктів.</p> <p>У результаті вивчення навчальної дисципліни студент повинен знати:</p> <ul style="list-style-type: none"> - фундаментальні принципи сучасного програмування; - сучасні засоби алгоритмічних мов програмування; - правила та властивості для побудови різнотипних алгоритмів; - основні правила і методи розроблення програм;
---------------------	---

	<ul style="list-style-type: none"> - основні структури даних та функцій; - основні принципи об'єктно-орієнтованого програмування. Вміти: - застосовувати фундаментальні основи та принципи сучасного програмування; - використовувати сучасні інструментальні засоби мов програмування; - визначати для вирішення задачі технологію та мову програмування; - здійснювати функціональну та об'єктну декомпозицію програми до обраної технології програмування; - самостійно виконувати розробку програмного коду програми; - налагоджувати та тестувати програму, виправляти синтаксичні та семантичні помилки; - володіти методами та технологіями об'єктно-орієнтованого програмування та програмування динамічних структур даних; - використовувати засоби сучасних мов програмування для створення програмних продуктів та застосовувати їх під час програмної реалізації алгоритмів професійних задач. <p>Дисципліна «Програмування» належить до переліку нормативних освітніх компонент освітньої програми «Комп'ютерна інженерія» та вибіркового освітнього компоненту для інших освітніх програм освітнього рівня «бакалавр».</p> <p>Силабус навчальної дисципліни «Програмування» складений відповідно до освітньо-професійної програми «Комп'ютерна інженерія» підготовки бакалаврів спеціальності 123 «Комп'ютерна інженерія».</p> <p><i>У випадку вибору курсу та його опанування достатня підготовка за програмою повної загальної середньої освіти</i></p>
Компетентності:	<p>Загальні компетентності</p> <p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК2. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>ЗК8. Здатність працювати в команді.</p> <p>Спеціальні (фахові) компетентності</p> <p>Р2. Здатність використовувати сучасні методи і</p>

	мови програмування для розроблення алгоритмічного та програмного забезпечення.
Програмні результати навчання:	<p>№6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.</p> <p>№8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.</p> <p>№12. Вміти ефективно працювати як індивідуально, так і у складі команди.</p> <p>№20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.</p>

3. Структура курсу

№	Тема	Результати навчання	Завдання
1.	Вступ у програмування. Прості типи, операції над ними. Умовний оператор та цикли	Знати базову структуру програм на мові C++, застосування та призначення мови програмування C++. Вміти формувати код програми та здійснювати її запуск. Знати базові типи мови C++. Вміти керувати вводом і виводом у C++. Знати загальні типи розгалужень та циклів у C++ та їх розширені можливості. Здійснювати операції над типами даних у C++.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
2.	Функції у C++. Контейнери.	Вміти створювати функції користувача для виконання окремих задач у програмі, передавати параметри функції за значенням, передавати параметри функцій за посиланням, передавати параметри функції за константним посиланням. Знати можливості та способи використання у програмах контейнерів <code>vector</code> , <code>map</code> , <code>set</code> .	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
3.	Алгоритми. Вступ у	Знати основні вбудовані алгоритми	

№	Тема	Результати навчання	Завдання
	структури і класи	та функції: min, max, sort, count, count_if, лямбда-функції. Розуміти видимість і способи ініціалізації змінних. Знати базові поняття створення структур і класів та їх призначення, конструкторів та деструкторів.	
4.	Класи. Робота з файлами і потоками. Винятки.	Знати принципи створення структур і класів, роботи з текстовими файлами і потоками, ефективногт використання потоків вводу і виводу. Вміти перевантажувати оператори вводу і виводу, +, <, сортування інтервалів. Вміти опрацьовувати виняткові ситуації у програмах.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
5.	Цілочисельні типи, кортежі, шаблонні функції.	Знати основні цілочисельні типи мови C++ та їх застосування, кортежі і пари. Вміти повертати кілька значень з функцій, створювати шаблони функцій.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
6.	Юніт-тести для налагодження програм	Вміти писати юніт-тести для відлагодження програм.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
7.	Файли заголовків. Багатофайлові проекти	Знати принципи побудови багатофайлових проектів, їхнє призначення та можливості	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
8.	Ітератори, алгоритми і контейнери	Знати, що таке ітератори, концепцію напівінтервалів, ітератори множин і словників, використання ітераторів в алгоритмах і контейнерах, відмінності між ітераторорами векторів і множин.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи
9.	Спадкоємство і поліморфізм	Розуміти можливості спадкоємства, забезпечувати необхідний захист та доступ до полів класів, знати списки ініціалізації. Вміти використовувати поліморфізм у програмному забезпеченні, створювати віртуальні методи, абстрактні класи.	Лабораторні роботи (звіт, програмний код), тестування під час кожної лабораторної роботи

4. Система оцінювання курсу

Накопичування балів під час вивчення дисципліни	
Види навчальної роботи	Максимальна кількість балів
Лекції	10

Лабораторні роботи	30
Самостійна робота	10
Індивідуальне завдання	-
Екзамен	50
Максимальна кількість балів	100

5. Оцінювання відповідно до графіку навчального процесу

Види навчальної роботи	Навчальні тижні																					Разом
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
Лекції (ваг. коеф. 0,05)								100									100					10
Лабораторні роботи (ваг. коеф. 0,7)	100	100	100	100	100	100			100	100	100	100	100	100	100	100		100	100			30
Самостійна робота							10										10					10
Індивідуальні завдання																						
Екзамен																						50
Всього за тиждень	100	100	100	100	100	100	10	100	100	100	100	100	100	100	100	100	110	100	100			100

6. Ресурсне забезпечення

Матеріально-технічне забезпечення	Мультимедійний проектор, комп'ютерна лабораторія для лабораторних робіт з IDE для програмування на С++ з кількістю комп'ютерів від 12 до 14, комп'ютерів
Література:	
<ol style="list-style-type: none"> 1. Запухляк Р.І. Програмування на С++. –Івано-Франківськ: ВДЦ Прикарпатського національного університету імені Василя Стефаника, 2009. -439 с. (мережа Інтернет). 2. Стивен Прата. Язык программирования С++ (С++11). Лекции и упражнения, 6-е издание — М.: Вильямс, 2012. — 1248 с. 3. Б. Страуструп Язык программирования С++. Специальное издание. Пер. с англ. — М.: Издательство Бином, 2011 г. — 1136 с. 4. Мейерс С. Эффективное использование С++. 35 новых рекомендаций по улучшению ваших программ и проектов. — М.: ДМК Пресс, 2014. 	

- 294с.
5. Мейерс С. Эффективное использование C++. 55 верных способов улучшить структуру и код ваших программ. — М.: ДМК Пресс, 2006. — 300с.
 6. Br. David Carlson. Software Design Using C++ (<https://cis.stvincent.edu/html/tutorials/swd/>)(інтернет).
 7. Pat Morin. Open Data Structures (in C++) (<https://docs.google.com/viewer?url=http%3A%2F%2Fopendatastructures.org%2Fods-cpp.pdf>)(інтернет).
 8. Скотт Мейерс. Эффективный и современный C++: 42 рекомендации по использованию C++11 и C++14. –Диалектика-Вильямс, 2020. – с. 304.
 9. Галовиц Я. C++17 STL. Стандартная библиотека шаблонов. — СПб.: Питер, 2018. — 432 с.
 10. Роберт С. Мартин. Гибкая разработка программ на Java и C++: принципы, паттерны и методики. –Издательство : Диалектика-Вильямс, 2020. –704 с.
 11. Страуструп Б. Дизайн и эволюция C++: Пер. с англ. – М.: ДМК Пресс, 2006. – 448 с.
 12. Энтони Уильямс. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ. Пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2012. – 672с.
 13. Бегун А.В. Технологія програмування: об'єктно-орієнтований підхід.- К.:КНЕУ,2000 .-200 с.(бібліотека 3 прим.).

7. Контактна інформація


Кафедра	Комп'ютерної інженерії та електроніки, вул. Шевченка, 57, 210а, 59-60-07, https://kkite.pnu.edu.ua/ , kkie@pnu.edu.ua
Викладач (і) Гостьові лектори	Запукхляк Руслан Ігорович, к.ф.-м.н., доцент
Контактна інформація викладача	ruslan.zapukhlyak@pnu.edu.ua

8. Політика навчальної дисципліни

Академічна доброчесність	Студент повинен бути толерантним і поважати думку інших. Заперечення повинні формулюватися тільки в коректній формі. Плагіат та академічна недоброчесність несумісні з принципами діяльності ЗВО. Не допускається підказування та
--------------------------	---

	<p>списування під час здачі будь-яких робіт поточного, рубіжного чи підсумкового контролю. Не допускається користування телефонами та будь-якими іншими електронними засобами під час здачі будь-яких робіт поточного, рубіжного, чи підсумкового контролю.</p> <p>За недотримання академічної доброчесності, студент може бути недопущений до складання підсумкового контролю та відрахований з університету.</p>
Пропуски занять (відпрацювання)	-1 бал за кожне заняття
Виконання завдання пізніше встановленого терміну	-20% балів від отриманого результату
Невідповідна поведінка під час заняття	-50% балів від отриманого результату
Додаткові бали	<p>Додається до 10 балів за вчасне виконання всіх видів контролю, відвідування всіх занять, належну поведінку. Якщо у підсумку кількість балів є більшою за 100 балів, то підсумкова оцінка встановлюється така, що дорівнює 100.</p>
Неформальна освіта	<p>Можливість зарахування. Рекомендовані платформи: Coursera, Prometheus.</p>

Викладач



Р.І. Запукхляк