

**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. Частина 2

Освітня програма **Професійна освіта. Цифрові технології**

Спеціалізація (за наявності) _____

Спеціальність **015 Професійна освіта (за спеціалізаціями)**

Галузь знань **01 Освіта/Педагогіка**

Затверджено на засіданні кафедри
Протокол № 12_від “30” червня 2023 р.

м. Івано-Франківськ – 2023 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	9

1. Загальна інформація

Назва дисципліни	Програмування. Частина 2.
Освітня програма	Професійна освіта. Цифрові технології
Спеціалізація (за наявності)	
Спеціальність	115 Професійна освіта (за спеціалізаціями)
Галузь знань	01 Освіта/Педагогіка
Освітній рівень	бакалавр
Статус дисципліни	Вибіркова
Курс / семестр	2 курс, 1 семестр
Розподіл за видами занять та годинами навчання (якщо передбачені інші види, додати)	Лекції – 20 год. Лабораторні заняття – 10 год. Самостійна робота – 60 год.
Мова викладання	Українська
Посилання на сайт дистанційного навчання	https://d-learn.pnu.edu.ua

2. Опис дисципліни

Мета та цілі курсу:	<p>Навчитися писати на C ++ ефективний код, вивчити ідіоми мови C ++, тобто показати, як за допомогою різних можливостей мови створювати елегантні, ефективні і надійні блоки коду. У сукупності зі знаннями, отриманими в основному курсі «Програмування», це дозволить студенту створювати не тільки швидкі і легко підтримувані програми. Крім того, будуть висвітлені теми, які не містилися в основному курсі «Програмування»: простори імен, розподіл пам'яті, багатопотоковість.</p> <p><i>Для вибору курсу та його опанування, необхідна підготовка, що відповідає знанням та вмінням з основного курсу «Алгоритмізація та програмування».</i></p>
Компетентності:	<p>Інтегральна компетентність: Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми в професійній освіті, що передбачає застосування певних теорій і методів педагогічної науки та інших наук відповідно до спеціалізації і характеризується комплексністю та невизначеністю умов.</p> <p>Загальні компетентності: ЗК 06. Навички використання інформаційних і комунікаційних технологій. ЗК 07. Здатність вчитися і оволодівати сучасними знаннями.</p>

	<p>Спеціальні (фахові компетентності):</p> <p>СК 08. Здатність використовувати відповідне програмне забезпечення для вирішення професійних завдань, відповідно до спеціалізації.</p> <p>СК 18. Здатність використовувати методології та технології проектування і застосування програмного забезпечення.</p> <p>СК 19. Здатність розробляти програмне забезпечення використовуючи методи та технології об'єктно-орієнтованого програмування.</p>
<p>Програмні результати навчання:</p>	<p>Знання:</p> <p>ПР 27. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p> <p>ПР 28. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p>

3. Структура курсу

№	Тема	Результати навчання	Завдання
1.	Макроси і шаблони класів.	Розуміти призначення макросів, знати випадки їх застосування та випадки, коли їх використовувати неприйнятно. Вміти програмувати макроси. Розуміти призначення шаблонів класів та навчитися їх створювати для розробки ефективного програмного забезпечення.	Бліц-питання на лекції, тести (макроси, вступ у шаблони класів, автоматичний вивід шаблонних типів), завдання для самостійної роботи
2.	Принципи оптимізації коду, складність алгоритмів і ефективне використання вводу/виводу	Знати правила оптимізації коду та ефективно їх застосовувати на практиці. Вміти ефективно використовувати потоки вводу та виводу. Вміти оцінювати складність алгоритмів.	Бліц-питання на лекції, тест (складність алгоритмів), завдання для самостійної роботи
3.	Модель пам'яті в C++	Розуміти як розподіляється пам'ять між різним об'єктами у C++, як її виділяти об'єкту, як звільняти виділену пам'ять. Навчитися ефективно керувати наявними ресурсами комп'ютерної системи у програмному коді.	Бліц-опитування під час викладу матеріалу, завдання, завдання для самостійної роботи
4.	Базова багатопотоковість	Вміти розпаралелювати задачі на потоки для ефективного використання сучасних процесорів у прикладному програмному забезпеченні.	Бліц-питання, тести, завдання для самостійної роботи

4. Система оцінювання курсу

Накопичування балів під час вивчення дисципліни	
Види навчальної роботи	Максимальна кількість балів
Лекції	10
Лабораторні роботи	70
Самостійна робота	20
Індивідуальне завдання	-
Залік/Екзамен (сума)	100
Максимальна кількість балів	100

5. Оцінювання відповідно до графіку навчального процесу

Види навчальної роботи	Порядковий номер заняття																		Разом			
	1	2	3	4	5	6	7	8	9	КСР	10	11	12	13	14	15	16	17		18	КСР	КСР
Лекції (ваг. кое ф. 0,05)										100											100	10
Лабораторні роботи (ваг. кое ф. 0,7)		100		100				100			100		100		100							70
Самостійна робота										10										10		20
Індивідуальні завдання																						
Залік (сума балів за всіма видами контролю)																						100

6. Ресурсне забезпечення

Матеріально-технічне забезпечення	Мультимедійний проектор, комп'ютерна лабораторія для лабораторних робіт з IDE для програмування на C++ з кількістю комп'ютерів від 12 до 14, комп'ютерів
Література:	
1. Запужляк Р.І. Програмування на C++. –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 С++. 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 С++: 42 рекомендации по использованию С++11 и С++14. –Диалектика-Вильямс, 2020. – с. 304.
 9. Галовиц Я. С++17 STL. Стандартная библиотека шаблонов. — СПб.: Питер, 2018. — 432 с.
 - 10.Роберт С. Мартин. Гибкая разработка программ на Java и С++: принципы, паттерны и методики. –Издательство : Диалектика-Вильямс, 2020. –704 с.
 - 11.Страуструп Б. Дизайн и эволюция С++: Пер. с англ. – М.: ДМК Пресс, 2006. – 448 с.
 - 12.Энтони Уильямс. Параллельное программирование на С++ 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.).

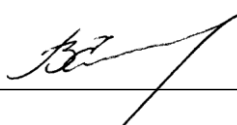
7. Контактна інформація

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 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. За недотримання академічної доброчесності, студент може бути недопущений до складання підсумкового контролю та відрахований з університету.
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	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.
Неформальна освіта	Можливість зарахування. Рекомендовані платформи: Coursera , Prometheus .

Викладач



Р.І. Запухляк