



Силабус освітнього компонента Програма навчальної дисципліни



Основи високопродуктивного програмного забезпечення

Шифр та назва спеціальності
122 – Комп'ютерні науки

Інститут
ННІ Комп'ютерного моделювання, прикладної фізики та математики

Освітня програма
Комп'ютерні науки. Моделювання, проектування та комп'ютерна графіка

Кафедра
Комп'ютерне моделювання процесів та систем (162)

Рівень освіти
Бакалавр

Тип дисципліни
Профільований пакет 1, Вибіркова

Семестр
8

Мова викладання
Українська

Викладачі, розробники



Сенько Альона Володимирівна

Alyona.Senko@khi.edu.ua

Доктор філософії, старший викладач

Основні наукові результати отримані у напрямку комп'ютерного моделювання процесів високотемпературного руйнування елементів конструкцій.

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Вивчаються концепції на високому рівні, системи володіння Rust, структури та методи, перелічені типи, вирази. Розглядається система модулів Rust та правила доступу для організації вашого коду та його публічного програмного інтерфейсу (API), деякі узагальнені структури даних - колекції, які надає стандартна бібліотека, такі як вектори, рядки. Досліджується філософія та техніка обробки помилок Rust. Вивчається тестування, яке є необхідним для забезпечення правильної логіки програми навіть за гарантій безпеки Rust. Досліджується замикання та ітератори - особливості Rust, які походять з функціональних мов програмування. Розглядається порівняння ідіом Rust із об'єктноорієнтованими принципами програмування.

Мета та цілі дисципліни

Сформувані у студентів знання, вміння і навички, необхідні для програмування завдань різної складності з використанням мови програмування Rust. Навчитись ефективно застосовувати теоретичний апарат та практичні навички створення програмних модулів для розробки програм.

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – іспит.

Компетентності

ЗК2: Здатність застосовувати знання у практичних ситуаціях

ЗК3: Знання та розуміння предметної області та розуміння професійної діяльності

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

СК10: Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника

Результати навчання

ПР1: Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук

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

Обсяг дисципліни

Загальний обсяг дисципліни 90 год. (3 кредитів ECTS): лекції – 20 год., лабораторні роботи – 20 год., самостійна робота – 50 год.

Передумови вивчення дисципліни (пререквізити)

Дисципліна базується на знаннях та компетенціях, що набуває здобувач вищої освіти під час вивчення дисциплін: Алгоритмізація та програмування, Об'єктно-орієнтоване програмування та проектування, Аналіз та тестування програмних систем

Особливості дисципліни, методи та технології навчання

Лекції проводяться інтерактивно з використанням мультимедійних технологій, де використовуються пояснювально-ілюстративний метод, метод критичного мислення та дискусії для викладу теорії та аналізу коду. На лабораторних роботах акцентується на практичному та частково-пошуковому методах, що сприяє розвитку практичних навичок програмування та вирішенню конкретних задач.

Навчальні матеріали доступні студентам на Microsoft OneDrive.

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Загальні концепції програмування. Знайомство з RUST.

Основи синтаксису Rust. Змінні та їх змінність. Типи даних. Функції. Коментарі. Управління потоком виконання.

Тема 2. Володіння - унікальна особливість мови Rust. Використання структур для структурування пов'язаних даних

Що таке володіння? Посилання і позичання. Тип даних слайс. Визначення та створення екземплярів структур Struct. Приклад програми, що використовує структури. Синтаксис методів

Тема 3. Перелічені типи і зіставлення з шаблоном. Керування проектами, що зростають, за допомогою пакетів, крейтів та модулів

Визначення перелічених типів. Конструкція управління match. Керування проектами, що зростають, за допомогою пакетів, крейтів та модулів. Пакети та крейти. Визначення модулів для управління областями видимості та приватністю. Шлях для доступу до елементів у дереві модулів.

Тема 4. Загальні колекції. Обробка помилок

Збереження списків значень у векторах. Збереження закодованого тексту UTF-8 у строках. Зберігання ключів зі зв'язаними значеннями у HashMap. Обробка помилок. Непереборні помилки та помилки, що піддаються відновленню за допомогою.

Тема 5. Узагальнені типи, типажі та час життя. Написання автоматизованих тестів.

Узагальнені типи даних. Трейти: визначення загальної поведінки. Перевірка коректності посилань за допомогою часів існування. Написання автоматизованих тестів. Як писати тести.

Тема 6. Проект з введенням/виведенням: створення програми командного рядка.

Приймання аргументів командного рядка. Читання файлу. Рефакторизація для покращення модульності та обробки помилок. Робота зі змінними середовища. Написання повідомлень про помилки в Standard Error замість стандартного виводу.

Тема 7. Функціональні можливості мови: ітератори та замикання. Більше про cargo та crates.io

Замикання: анонімні функції, що захоплюють своє середовище. Обробка послідовностей елементів за допомогою ітераторів. Покращуємо наш проект з введенням/виведенням. Більше про cargo та crates.io Налаштування збірок з release профілями. Робочі області cargo.

Тема 8. Розумні вказівники. Конкурентність без страху

Використання розумних вказівників як звичайних посилань. Виконання коду при очищенні за допомогою трейту Drop. Розумний вказівник з лічильником посилань.

Тема 9. Можливості об'єктно-орієнтованого програмування Rust. Шаблиони.

Характеристики об'єктно-орієнтованих мов. Реалізація шаблону об'єктно-орієнтованого проектування. Де можуть використовуватися шаблони? Синтаксис шаблон

Тема 10. Просунутий функціонал. Останній проект: збірка багатопотокового вебсервера

Просунутий функціонал. Небезпечний Rust. Поглиблено про трейти. Поглиблено про типи. Поглиблено про функції та замикання. Останній проект: збірка багатопотокового вебсервера. Збірка однопотокового вебсервера. Перетворюємо наш однопотоковий сервер на багатопотоковий. Плавне вимикання і очищення.

Теми практичних занять

Не передбачено навчальним планом.

Теми лабораторних робіт

Тема 1. Знайомство з RUST.

Тема 2. Використання структур.

Тема 3-4. Колекції: Vectors, Strings, Hash Maps.

Тема 5. Обробка помилок.

Тема 6. Написання автоматизованих тестів.

Тема 7. Проект з введенням/виведенням: створення програми командного рядка.

Тема 8. Об'єктно-орієнтоване програмування Rust.

Тема 9-10. Збірка багатопотокового веб-сервера.

Самостійна робота

Теми самостійної роботи:

1 Лаконічний контроль виконання конструкцією if let.

2 Підключення шляхів до області видимості за допомогою ключового слова use.

3 Розподіл модулів на різні файли.

4 Непереборні помилки з макросом panic.

5 Помилки, що піддаються відновленню за допомогою Result.

6 Контроль над запуском тестів.

7 Організація тестів.

8 Порівняння швидкодії: цикли проти ітераторів.

9 Розширення cargo із користувацькими командами.

10 Використання трейт-об'єктів, що допускають значення різних типів.

Література та навчальні матеріали

1. Klabnik Steve The Rust Programming Language: 2nd Edition / Steve Klabnik. 2023. - 560
2. Blandy Jim Programming Rust. Fast, Safe Systems Development. Editie: 2nd Revised edition / Jim Blandy. 2021. - 622
3. Metzler Nathan Rust Programming for Beginners: An Introduction to Learning Rust Programming with Tutorials and Hands-On Examples / Nathan Metzler. 2021. - 174
4. Flitton Maxwell Rust Web Programming: A hands-on guide to developing fast and secure web apps with the Rust programming language / Maxwell Flitton .2021 - 394
5. Abella Hernando Rust for Beginners: Let's Learn together to code in Rust. / Hernando Abella. 2023 - 111

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

Бали нараховуються за наступним співвідношенням:

- складання всіх лабораторних робіт: 60 балів
- Іспит складається з двох частин:
- теоретично-практична частина іспиту: від 0 до 30 балів
 - програмна частина іспиту (після успішного завершення 1-ї частини та отримання мінімум 15 балів): від 0 до 10 балів

Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту. Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Погодження

Силабус погоджено

28.08.2023

Завідувач кафедри
Дмитро БРЕСЛАВСЬКИЙ

28.08.2023

Гарант ОП
Оксана ТАТАРІНОВА