



Силабус освітнього компонента

Програма навчальної дисципліни



Алгоритмізація та програмування

Шифр та назва спеціальності

122 – Комп'ютерні науки

Інститут

ННІ Комп'ютерних наук та інформаційних технологій

Освітня програма

Комп'ютерні науки та інтелектуальні системи

Кафедра

Програмна інженерія та інтелектуальні технології управління (321)

Рівень освіти

Бакалавр

Тип дисципліни

Спеціальна (фахова), Обов'язкова

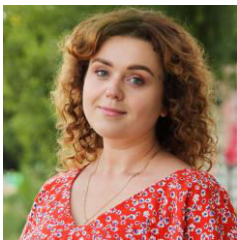
Семестр

1, 2

Мова викладання

Українська, англійська

Викладачі, розробники



Білова Марія Олексіївна

mariia.bilova@khipi.edu.ua

Кандидат технічних наук, доцент, доцент кафедри ПІІТУ НТУ "ХПІ"

Підготувала і опублікувала понад 50 публікацій (Google Scholar: <https://scholar.google.com/citations?user=b3YLGToAAAAJ>; ORCID: <https://orcid.org/0000-0001-7002-4698>; Scopus <https://www.scopus.com/authid/detail.uri?authorId=57190442390>).
[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Курс охоплює широкий спектр тем, пов'язаних з програмуванням, алгоритмами, структурами даних та комп'ютерними науками загалом та сприяє розвитку аналітичних навичок, критичного мислення та творчого підходу до розв'язання проблем. Студенти вивчають основні концепції програмування мовою C, а також знайомляться з мовами вищого рівня, такими як Python та JavaScript.

Під час навчання ви отримаєте практичні навички шляхом вирішення завдань і розробки власних проектів.

В основу курсу покладено CS50 - відомий вступний курс з комп'ютерних наук, що викладається Гарвардським університетом. Студенти, що успішно впораються з усіма завданнями, зможуть отримати сертифікат про проходження курсу, завірений гарвардським викладачем курсу, професором Девідом Маланом.

Мета та цілі дисципліни

Формування у студентів алгоритмічного мислення та набуття навичок розробки програм на мовах програмування C та Python для вирішенні прикладних задач із різних предметних областей відповідно до сучасних принципів і тенденцій побудови прикладного програмного забезпечення.

Формат занять

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

Компетентності

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.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК9. Здатність працювати в команді.

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.

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 330 год. (11 кредитів ECTS): лекції – 90 год., лабораторні роботи – 90 год., самостійна робота – 150 год.

Передумови вивчення дисципліни (пререквізити)

Базове розуміння математичних концепцій, таких як логіка, алгебра та статистика, а також базові навички роботи з комп'ютером

Особливості дисципліни, методи та технології навчання

Методи викладання та навчання:

інтерактивні лекції з презентаціями, дискусії, лабораторні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання.

Форми оцінювання:

письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування (CAS), підсумковий/семестровий контроль у формі семестрового екзамену, відповідно до графіку навчального процесу (FAS).

Програма навчальної дисципліни

Теми лекційних занять

Семестр 1

Тема 1. Алгоритми та особливості їх розробки

Мета вивчення та особливості курсу. Поняття чорного ящика. Алгоритми та алгоритмічне мислення. Псевдокод. Графічне представлення алгоритму. Діаграма діяльності.

Тема 2. Системи числення

Способи представлення даних. Поняття системи числення. Десяткова система. Двійкова система. Вісімкова система. Шістнадцяткова система. Перехід між системами. Таблиця ASCII. Колірні моделі. Основи мови програмування SCRATCH.

Тема 3. Типи даних та змінні в мові C

Поняття типу даних. Особливості типізації мови C. Visual Studio Code. Типи даних в мові C. Поняття змінної. Особливості іменування змінних в мові C. Оголошення та ініціалізація змінних. Области видимості змінної. Оператори. Оператори присвоєння. Математичні оператори. Логічні оператори. Оператори порівняння.

Тема 4. Умовні оператори

Розгалуження у програмі. Оператор if. Оператор if else. Оператор if else if else. Оператор switch. Тернарний оператор.

Тема 5. Цикли

Поняття циклу. Особливості використання циклів в мові C. Цикл for. Цикл while. Цикл do while.

Тема 6. Масиви

Поняття масиву. Оголошення та ініціалізація масивів. Основні характеристики масивів. Індексція в масивах. Рядок як масив символів. Багатовимірні масиви. Особливості роботи з багатовимірними масивами.

Тема 7. Функції

Поняття функції. Оголошення та визначення функції. Прототип функції. Виклик функції. Формальні та фактичні параметри функції. Оператор return. Функція main. Аргументи командного рядка. Поняття рекурсії. Складові рекурсії

Тема 8. Особливості роботи з пам'яттю

Поняття стеку та купи. Вказівники. Динамічне виділення пам'яті в C. Використання sizeof. Очищення пам'яті. Зміна розміру виділеної пам'яті. Поняття файлу. Особливості взаємодії з файлами в C. Поняття потоку.

Тема 9. Структури

Поняття структури. Створення структури. Особливості використання структур в мові C. Структури даних Поняття однозв'язного, двузв'язного списку. Асоціативний масив. Хеш таблиці та хеш-функції. Розв'язання колізій. Особливості реалізації структур даних в C.

Семестр 2

Тема 1. Мова програмування Python. Типи даних

Особливості типізації. Типи даних. Робота зі змінними в Python. Оператори: математичні, логічні, оператори присвоєння, оператори приналежності, оператори порівняння.

Тема 2. Умови та цикли у Python

Особливості умовних конструкцій у Python. Цикл while. Цикл for.

Тема 3. Функції та особливості роботи з рядками у Python

Функції. Аргументи функції. Рядки у Python. Індексція в рядках, від'ємна індексція. Робота з підмножиною рядка.

Тема 4. Основи баз даних

Дані. Бази даних. Системи управління базами даних. SQLite. Реляційні бази даних. Основи проектування баз даних.

Тема 5. Основи мови SQL

Зберігання даних у файлах CSV. Основні команди DDL та DML.

Тема 6. Основи UML

Уніфікована мова моделювання. Інструменти для розробки діаграм у нотації UML. Діаграма варіантів використання. Діаграма класів. Діаграма розгортання. Діаграма послідовності.

Тема 7. Основи веб-розробки

Мова розмітки гіпертексту. Структура веб-сторінки. Document Object Model. Блочні та рядкові теги. Каскадні таблиці стилів. Правила CSS. Різновиди селекторів. CSS-властивості.

Тема 8. Основи backend розробки

Протокол HTTP. Фреймворк Flask. Найпростіший застосунок Flask. Маршрутизація. Візуалізація шаблонів. Робота з базами даних у застосунку.

Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

Теми лабораторних робіт

Семестр 1

Тема 0. Scratch

Тема 1. Основні синтаксичні конструкції мови програмування C

Тема 2. Масиви та функції

Тема 3. Алгоритми

Тема 4. Робота з пам'яттю

Тема 5. Структури даних

Семестр 2

Тема 1. Основні концепції мови програмування Python

Тема 2. Основи SQL

Тема 3. Основи HTML та CSS

Тема 4. Створення застосунку на основі Flask

Самостійна робота

Семестр 1

В рамках першого семестру студентам пропонується виконати 2 індивідуальних домашніх завдання:

- 1) розробка діаграми діяльності відповідно варіанту (5 балів);
- 2) розробка програми на основі діаграми діяльності з ІДЗ 1 (5 балів).

Семестр 2

При в рамках другого семестру студенти повинні виконати курсову роботу відповідно до обраної теми. Курсова робота має бути написана з використанням HTML, CSS, Python та мікрофреймворку Flask, а також передбачає проектування та розробку бази даних. Курсова робота складає 30% від оцінки другого семестру, з яких 15 балів нараховуються за проведення дослідження, проектування та виконання супровідної документації у вигляді записки, а 15 - за проект та його презентацію (захист курсової роботи). Можливі теми курсової роботи подані нижче.

Розробка програмного забезпечення для тренування реакції користувача

Розробка програмного забезпечення для генерації палеток кольору на основі зображення

Розробка програмного забезпечення для тренування пам'яті користувача

Розробка програмного забезпечення для відстеження ваги користувача

Розробка програмного забезпечення для тестування знань студента про обрану мову програмування

Розробка програмного забезпечення для підбору випадкових фільмів відповідно до настрою користувача

Розробка програмного забезпечення для організації робочого часу користувача (трекер задач)

Розробка програмного забезпечення для фіксації даних про прочитані книжки

Розробка програмного забезпечення для відстеження настрою користувача

Розробка програмного забезпечення для відстеження витрат користувача

Розробка програмного забезпечення для відстеження ігрового часу користувача

Розробка програмного забезпечення для планування візитів до лікарів

Розробка програмного забезпечення про розділення відходів з відображенням карти місць, де можна здати сміття

Розробка програмного забезпечення для розвитку уважності користувача

Розробка програмного забезпечення для зберігання та відслідковування поставлених цілей користувача

Розробка програмного забезпечення для вивчення правописання нових слів

Розробка програмного забезпечення для відстеження даних стосовно здоров'я домашнього улюбленця

Розробка програмного забезпечення для відображення країн, які відвідав користувач

Розробка програмного забезпечення для розрахунку калорійності раціону людини

Розробка програмного забезпечення для організації процесу обміну книжками

Розробка програми для фіксації обсягу випитої користувачем води з відображенням статистики
Розробка програмного забезпечення для накладання фільтрів на зображення
Розробка програмного забезпечення для відстеження занять в тренажерному залі
Розробка програмного забезпечення для підрахунку сумісності за знаком зодіаку
Розробка програмного забезпечення для поглиблення знань стосовно форм дієслів в англійській мові
Розробка програмного забезпечення для відстеження успішності користувача при навчанні в автомобільній школі
Можлива інша тема за бажанням студента, узгоджена з викладачем

Література та навчальні матеріали

Базова література

1. Корнієнко М. М., Іванова І. Д. Інформатика. Основи алгоритмізації і програмування. – Ранок, 2011. – 48 с.
2. Cormen T. H. Introduction to Algorithms Third Edition / Thomas H. Cormen Charles E. Leiserson Ronald L. Rivest Clifford Stein // The MIT Press, 2009. – 1313 p.
3. Stephens R. Essential Algorithms: A Practical Approach to Computer Algorithms. – John Wiley & Sons, Inc, 2013. – 544 с.
4. C Programming Absolute Beginner's Guide. Third Edition. - Pearson Education, 2014. – 617с.
5. Seacord R. C. Effective C: An Introduction to Professional C Programming. – No Starch Press, 2020. – 272 p.
6. Thomas Mailund Pointers in C Programming. A Modern Approach to Memory Management, Recursive Data Structures, Strings, and Arrays. – Apress,
7. Anquetil R Fundamental Concepts for Web Development: HTML5, CSS3, JavaScript and much more. - Independently published, 2019. – 276 с.
8. Myers M. Smarter way to learn Python. – 2017. – 234 p.
9. Stephenson B. The Python Workbook. – Springer: Texts in Computer Science, 2019. – 218 p.
10. Маттес Е. Пришвидшений курс Python. – Видавництво Старого Лева, 2021. – 600 с.
11. Руденко В.Д., Жугастров О.О. Основи алгоритмізації і програмування мовою Python. – Ранок, 2019. – 192 с.
12. Allen G. Taylor Author of SQL All-in-One For Dummies. 9th edition. Hoboken, 2019. – 496 p.
13. Upadhyay K. Ch. HTML5 For Web Designers. Complete Hypertext Markup Language Guidance. - Independently published, 2020. – 71 p.
14. Grant K. J. CSS in depth. – Manning Publications Co, 2018. – 445 p.
15. Grinberg M. Flask Web Development: developing web applications with Python. 2nd edition. - O'Reilly Media, Inc, 2018. – 314 p.

Допоміжна література

16. Skiena S. S. The Algorithm Design Manual. Third edition. – Springer, Texts in Computer Science, 2020. – 810 p.
17. Al Sweigart Invent Your Own Computer Games with Python, 4th edition. – No Starch Press, 2017. – 376 p.
18. Мартін Р. С. Чистий код. – 2019. – 368 с.
19. Chacon S. Pro Git [Electronic resource] / Scott Chacon, Ben Straub. Apress, 2014.– 608 p. – Mode of access: <https://git-scm.com/book/uk/v2>

Інформаційні ресурси в інтернеті

- The GNU C Reference Manual [Electronic resource]. – Access mode: <https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html>
21. Beej's Guide to C Programming [Electronic resource]. – Access mode: <https://beej.us/guide/bgc/html//index.html>
22. CS50 «Основи програмування» на платформі Prometheus [Електронний ресурс]. – Режим доступу: https://edx.prometheus.org.ua/courses/Prometheus/CS50/2016_T1/info
23. CS50 [Електронний ресурс]. – Режим доступу: <https://cs50.harvard.edu/>

24. C Programming Language Documentation. [Electronic resource]. – Access mode: <https://devdocs.io/c/>
25. Front-End Developer Handbook 2018 / Cody Lindley – Frontend Masters. –2018. – 168 p. [Electronic resource]. – Access mode : <https://legacy.gitbook.com/book/frontendmasters/front-end-developer-handbook-2018/details>.
26. SQL Tutorial [Electronic resource]. – Access mode : <https://www.w3schools.com/sql/>
27. CSS Snapshot 2017. W3C Working Group Note [Electronic resource]. – Access mode : <https://www.w3.org/TR/css-2017/>.
28. HTML 5.2. W3C Recommendation [Electronic resource]. – Access mode : <https://www.w3.org/TR/html52/>.
29. UML [Electronic resource]. – Access mode : <https://www.uml.org/>
30. Основи UML [Electronic resource]. – Access mode : <https://docs.kde.org/trunk5/uk/umbrello/umbrello/uml-basics.html>

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

Семестр 1

Підсумкова оцінка складається з оцінок за шість лабораторних робіт загальною сумою 48 бал, (Лабораторні роботи 0, 5 - по 10 балів, Лабораторні роботи 1-4 - по 7 балів), двох індивідуальних домашніх завдань загальною сумою в 10 балів (по 5 балів кожне), чотирьох контрольних робіт у формі тестів загальною сумою 12 балів (контрольні роботи 1-3 по 3 бали, контрольна робота 4 - 5 балів), а також екзамену - 30 балів. Основою до допуску до екзамену (обов'язковим навантаженням) є виконання лабораторних робіт 0-4.

Семестр 2

Підсумкова оцінка складається з оцінок за чотири лабораторні роботи загальною сумою 30 балів, (Лабораторні роботи 1-2 - по 7 балів, Лабораторні роботи 3-4 - по 8 балів), двох контрольних робіт у формі тестів загальною сумою 10 балів (по 5 балів кожна), курсової роботи, 30 балів, а також екзамену - 30 балів. Основою до допуску до екзамену (обов'язковим навантаженням) є виконання усіх лабораторних робіт, а також курсової роботи.

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

08.06.2023

Завідувач кафедри
Ігор ГАМАЮН

08.06.2023

Гарант ОП
Андрій КОПП