Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра автоматизації проектування енергетичних процесів і систем

ФУНКЦІОНАЛЬНЕ ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем і веб-технологій
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)
Рік підготовки, семестр	3 курс весінній семестр
Обсяг дисципліни	На засвоєння дисципліни передбачено 120 год / 4 кредитів ЄКТС, з них - 36 год. лекції, 18 год. практичні, 66 год. самостійна робота
Семестровий контроль/ контрольні заходи	Залік, МКР
Розклад занять	*http://rozklad.kpi.ua/*
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: д.т.н., доцент, Ходаковський Олексій Володимирович, dzgeron@ukr.net, тел. 050-825-33-53 Практичні: д.т.н., доцент, Ходаковський Олексій Володимирович, dzgeron@ukr.net, тел. 050-825-33-53
Розміщення курсу	Кампус

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчання та результати навчання

Предметом вивчення навчальної дисципліни є загальні принципи функціонального програмування, базові термінологія та навички програмування на функціональних мовах програмування, класичні
задачі та алгоритми їх розв'язання із використанням функціонального підходу у програмуванні, оволодіння технологією розробки програм на мові програмування Haskell.
Основними завданнями вивчення дисципліни «Функціональне програмування» є ознайомлення студентів із функціональним підходом до розробки прикладних програм, з існуючими технологіями функціонального програмування, окреслення ряду прикладних задач, які розв'язуються із використанням функціонального підходу, вивчення функціональної мови програмування Haskell.

Метою викладання навчальної дисципліни «Функціональне програмування» є засвоєння студентами основних концепцій, принципів та понять сучасного функціонального програмування, що створюють основу теоретичних досліджень і практичних розробок в області функціональних мов програмування (Haskell).

Предмет дисципліни – вивчення підходів, опанування математичним апаратом до процесу прийняття рішень у різних ризику і невизначеності.

Завдання. В результаті вивчення дисципліни у студентів повинні сформуватися наступні компетентності:

загальні:

- здатність до абстрактного мислення, аналізу та синтезу (ЗК 1),
- здатність спілкуватися іноземною мовою як усно, так і письмово (ЗК 4),
- здатність діяти соціально відповідально та свідомо (ЗК 10).

фахові:

- здатність розробляти архітектури, модулі та компоненти програмних систем (ФК 3),
- здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки) (ФК 6),
- здатність накопичувати, обробляти та систематизувати професійні знання щодо створення тестування і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя (ФК 10),
- здатність обґрунтовано обирати та освоювати інструментарій з розробки тестування та супроводження програмного забезпечення (ФК 13),
- здатність проектувати кібер-фізичні системи, володіти скриптовими та декларативними мовами програмування (ФК 16),
- здатність розробляти програмні системи з мікросервісною архітектурою, конструювати мобільні додатки, крос- та мульти-платформне програмування, зокрема, для кібер-фізичних систем (ФК 20).

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі програмні результати навчання:

- дотримуватися морально-етичних та культурних норм, принципів академічної доброчесності та кодексу професійної етики, примножувати досягнення суспільства (ПРН 1),
- знати професійні стандарти та інші нормативно- правові документи в галузі інженерії програмного забезпечення (ПРН 10),
- використовувати методологію створення програмного забезпечення згідно поставленої задачі. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення (ПРН 16),
- володіти навичками командної розробки, погодження, оформлення і випуску всіх видів програмної документації (ПРН 17).

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Пререквізити дисципліни. Основи теорії множин та елементи математичної логіки, що вивчаються в «Комп'ютерній дискретній математиці», курс «Теорія прийняття рішень», многочлени від однієї змінної, що вивчаються в «Прикладній математиці в ІТ-галузі», елементарна математика в обсязі програми загальноосвітньої школи.

Постреквізити дисципліни. Безпосереднє застосування результатів навчання при вивченні дисциплін «Програмування скриптовими мовами», «Безпека web-ресурсів та додатків», а також усіх тих дисциплін, які використовують результати навчання вищеназваних..

3. Зміст навчальної дисципліни

Тема 1 Вступ до функціонального програмування. Основні мови функціонального програмування Специфіка та основні параметри мови Haskell.
Тема 2 Типи даних. Управляючі конструкції.
Тема 3 Поняття про структурно-оптимальне програмування
Тема 4 Керуючі структури.
Тема 5 Експертні системи

Тема 6 Засоби об’єктно-орієнтованого програмування.
Тема 7. Самонавчаючи системи.
Тема 8. Особливості програмування різними мовами функціонального програмування.
Тема 9. Функції вищих порядків.
Тема 10. Модулі та абстрактні типи даних у мові Haskell.

4. Навчальні матеріали та ресурси

Основна література

1. Заяць В. М. Логічне і функціональне програмування. Системний підхід : підручник / В. М. Заяць, М. М. Заяць. – Рівне : НУВГП, 2018. – 422 с
2. Ліпован М. Вивчить собі Хаскела на велике щастя! / М. Ліповача. пер. Лелів Г. - 2017.
3. Нікітченко М.С. Теоретичні основи програмування: [навчальний посібник] / М. С. Нікітченко. – Ніжин: Видавництво НДУ імені Миколи Гоголя, 2010. – 211 с.
4. Шевченко І.В. Функціональне та логічне програмування (Частина 1. Функціональне програмування) / І.В. Шевченко, Ю.А. Кузнецова, М.О. Сьомочкін–Навч. посібник з виконання лабораторних робіт. – Харків: Нац. аерокосм. ун-т «Харк.авіац. ін-т », 2020. –92с.

Додаткова література

5. Штучний інтелект в засобах мультимедіа : Лаб. практикум /укладачі С. В. Кизима, Д. П. Кучеров. – К., НАУ, 2014. – 48 с

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Лекція 1. Вступ до функціонального програмування. Основні мови функціонального програмування Специфіка та основні параметри мови Haskell.

Історія виникнення функціонального програмування та його розвитку. Основна ідея функціонального програмування. Найбільш поширені мови функціонального програмування. Відміни від процедурного програмування. Зв’язування змінних.

Лекція 2. Типи даних. Управляючі конструкції.

Основні типи даних. Типи даних в мові Haskell. Складові частини символу та його внутрішнє представлення. Застосування властивостей символу.

Лекція 3. Поняття про структурно-оптимальне програмування

Застосування функцій IF, COND, DO, LOOP, PROG, Основні функції. Поіменована функція та її застосування. Базисні функції. Функції роботи із списками. Робота з контекстом. Функції вводу-виводу. Розгалуження розрахунків. Циклічні розрахунки. Розробка рекурсивних функцій.

Лекція 4. Керуючі структури

Керуючі структури та їх розробка. Функціонали та їх застосування. Макроси та їх застосування.

Лекція 5. Експертні системи

Засоби розробки експертних систем мовою Haskell. Розробка об’єктно-орієнтованої системи.

Лекція 6. Засоби об’єктно-орієнтованого програмування

Порівняння створення програмних засобів у різних мовах функціонального програмування.

Лекція 7. Самонавчаючи системи.

Використання параметрів нагромадження в функціональному програмуванні. Типи перерахувань. Типи добутків і сум добутків. Типи з параметрами. Рекурсивні типи даних. Синоніми та обгортки.

Лекція 8 Особливості програмування різними мовами функціонального програмування
Параметричний та спеціальний поліморфізм. Класи типів. Система модулів

Лекція 9. Функції вищих порядків
Формалізація задач прийняття рішень в умовах ризику та невизначеності. Критерії прийняття рішень Байєса-Лапласа, Севіджа, Гурвіца, Вальда, їх властивості та модифікації.

Лекція 10. Модулі та абстрактні типи даних у мові Haskell.

Поняття нечітких множини та функцій. Операції над нечіткими множинами. Поняття нечіткого бінарного відношення. Постановка задачі прийняття рішення з нечітко визначеною метою і обмеженнями.

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Відвідування лекційних та лабораторних занять є обов'язковим за винятком поважних причин (хвороби, форс-мажорних обставин).

В разі пропущення занять з поважних причин викладач надає можливість студенту виконати усі або деякі лабораторні завдання (винятком є виконання деяких завдань у зв'язку із закінченням навчального процесу).

В разі пропущення занять без поважних причин, а також через порушення граничного терміну виконання завдання (deadline) студент може отримати 80% від максимальної оцінки відповідне завдання.

Протягом семестру студенти:

- виконують та захищають лабораторні роботи у відповідні терміни (на кожну лабораторну роботу відводиться два тижні для здачі),
- пишуть модульну контрольну роботу,
- повинні позитивно закрити дві атестації (в кінці березня та в середині травня),
- по закінченні навчального процесу складають залік.

7. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Система рейтингових (вагових) балів та критерії оцінювання

1) Робота на лекціях

На лекціях може бути проведено бліцопитування студентів. Такі опитування проводяться на довільних лекціях 5 разів протягом семестру, наприкінці лекції. Ваговий бал за вірну відповідь - 1. Максимальна кількість балів, що може отримати кожен студент за семестр - 5.

2) Лабораторні практикуми

Максимальна кількість балів за усі виконані комп'ютерні практикуми дорівнює 50 балів. Розподіл балів серед лабораторних практикумів наступний:

№ з/п	Назва лабораторного практикуму	Кількість балів
1	Застосування функцій SET, QUOTE і SETQ, IF, COND	5
2	Застосування властивостей символу	5
3	Застосування функцій роботи із списками	5
4	Застосування рекурсивних функцій	5
5	Розробка експертної системи пошуку маршрута	5

6	Розробка спеціальних функцій за допомогою методу програмування, що управляється даними	5
7	Розробка об'єктно-орієнтованої системи	5
8	Розробка спеціальних функцій за допомогою методу програмування, що управляється подіями	10
Всього:		45

Критерії оцінювання:

Виконання лабораторного практикуму:

- виконаний своєчасно (протягом двох тижнів з моменту видачі), у повному обсязі – відповідний бал згідно номеру комп'ютерного практикуму;
- виконаний із запізненням – знімається 10 – 30% від максимальної кількості балів в залежності від терміну запізнення;
- виконаний не самостійно, із запізненням – знімається 50% від максимальної кількості балів;
- невиконани1 протягом відведеного часу – 0 балів.

3) Модульна контрольна робота

Максимальна кількість балів за модульну контрольну роботу дорівнює 10 балів.

Якість виконання роботи:

- усі відповіді вірні та повні –10 балів,
- у відповідях допущені несуттєві неточності – 8 балів,
- половина відповідей вірна – 5 балів,
- відповіді з суттєвими неточностями, але без критичних помилок – 2 бали,
- менше половини відповідей вірна – 0 балів.

Штрафні та заохочувальні бали за:

- активність на комп'ютерних практикумах + 2 бали
- виконання комп'ютерного практикуму з використанням власного оптимального алгоритму + 1 бали
- відсутність на занятті без поважної причини – 2 бали
- несвоєчасна здача комп'ютерного практикуму (пізніше ніж за тиждень) – 0,5 балів;

4) Складання заліку

Максимальний ваговий бал $r_{icп}$=40

На іспиті студент виконує письмову контрольну роботу, яка містить два теоретичних питання і одне практичне питання. Теоретичні питання оцінюються максимально по 10 балів, практичне – 20 балів.

Умови позитивної проміжної атестації

Для отримання „зараховано" з першої проміжної атестації студент матиме не менше ніж 11 балів (за умови, що за 8 тижнів згідно з календарним планом контрольних заходів „ідеальний" студент має отримати 3+4+4+5+5 = 21 бал).

Для отримання „зараховано" з другої проміжної атестації студент матиме не менше ніж 22 балів (за умови, що за 14 тижнів згідно з календарним планом контрольних заходів „ідеальний" студент має отримати 21 + 4 +5 +5 +10 =45 балів).

Умови допуску до *іспиту*

Необхідною умовою допуску до іспиту є зарахування усіх комп'ютерних практикумів та виконання модульної контрольної роботи, а також стартовий рейтинг (R_c) не менше 40 балів.

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру (шкала рейтингу) складає:

$$R = r_{лек} + r_{практ} + r_{мод} + r_{icп} = 5+45 + 10 + 40= 100 \text{ балів.}$$

Максимальний стартовий рейтинг становить $R_c = r_{лек} + r_{практ} + r_{мод} = 60$ балів.

Рейтинг заліку дорівнює 40 балів. Мінімальний рейтинг допуску до іспиту становить 40 балів.

Таким чином, рейтингова шкала з кредитного модуля складає

$$R = 60+40=100 \text{ балів.}$$

Для отримання студентом відповідних оцінок рейтингова оцінка студента переводиться згідно таблиці:

Бали	**Оцінка**
95 - 100	Відмінно
85 - 94	Дуже добре
75 - 84	Добре
65 - 74	Задовільно
60 - 64	Достатньо
Менше 60	Незадовільно
R < 40 є незараховані роботи комп'ютерного практикуму або не виконані інші умови допуску до екзамену	Не допущено

Робочу програму навчальної дисципліни (силабус):

Складено доцент, к.т.н., доцент, Ходаковський Олексій Володимирович

Ухвалено кафедрою АПЕПС (протокол № _16__ від _18.06.2021 р._)

Погоджено Методичною комісією ТЕФ КПІ ім. Ігоря Сікорського [1] (протокол № 11 від _24.06.2021 р.)

[1] Методичною радою університету – для загальноуніверситетських дисциплін.