

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«___» _____ 2018 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Інструментальні засоби та технології програмування

для студентів

галузь знань	12 «Інформаційні технології» <i>(шифр і назва)</i>
спеціальність	122 «Комп'ютерні науки» <i>(шифр і назва спеціальності)</i>
освітній рівень	магістр <i>(молодший бакалавр, бакалавр, магістр)</i>
освітня програма	«Інформатика» <i>(назва освітньої програми)</i>
вид дисципліни	вибіркова

Форма навчання	денна
Навчальний рік	2018/2019
Семестр	3
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	залік

Викладачі: д.ф.-м.н., проф. Буй Д.Б. (семінари)

Пролонговано: на 20__/20__ н.р. _____ (_____) «__» 20__ р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____ (_____) «__» 20__ р.
(підпис, ПІБ, дата)

КИЇВ – 2018

Розробник: Буй Дмитро Борисович, д.ф.-м.н., професор кафедри «Теорії та технології програмування»

ЗАТВЕРДЖЕНО

В.о. Зав. кафедри «Теорії та технології програмування»

_____ Панченко Т.В.
(підпис) (прізвище та ініціали)

Протокол № ____ від «____» _____ 20__ р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «____» _____ 20__ року № ____

Голова науково-методичної комісії _____ Хусаїнов Д.Я.
(підпис) (прізвище та ініціали)

Затверджено Вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «____» _____ 20__ року № ____

Голова Вченої ради _____ Анісімов А.В.
(підпис) (прізвище та ініціали)

ВСТУП

1. Мета дисципліни – засвоєння базових знань щодо використання різних парадигм програмування. Оволодіння навичками проектування та розробки програмних систем із застосуванням різноманітних парадигм програмування з відповідними структурами даних, механізмами управління та технологіями.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. Знати: загальні поняття мов програмування (синтаксис, семантика програми, об'єктно-орієнтований стиль програмування), основи теорії алгоритмів (рекурсивні функції та рекурсивні обчислення), основні поняття математичної логіки, зокрема, поняття резольвенти, резолютивного виводу.

2. Вміти: використовувати різноманітні середовища розробки, редагування та виконання програм.

3. Володіти елементарними навичками: програмування чи специфікації програм, що ґрунтуються на рекурсивних обчисленнях.

3. Анотація навчальної дисципліни:

Навчальна дисципліна "Інструментальні засоби та технології програмування" є складовою освітньо-наукової програми підготовки фахівців за *освітнім рівнем* «магістр» *галузі знань* 12 "Інформаційні технології" за *спеціальністю* 122 „Комп’ютерні науки”, програми «Інформатика».

Дана дисципліна є дисципліною спеціалізації «Теорія та технологія програмування» за освітньою програмою **“Інформатика”**.

Викладається у 3 семестрі в **обсязі – 150 годин.**

(5 кредитів ECTS) зокрема: *семінарські заняття – 44 год., самостійна робота – 106 год.* Завершується дисципліна **заліком в 3 семестрі.**

В результаті вивчення навчальної дисципліни студент повинен:

знати: суть і підґрунтя парадигми логічного програмування та парадигми функціонального програмування;

вміти: застосовувати логічний та функціональний стилі програмування при розв’язуванні програмістських задач.

Місце дисципліни. Навчальна дисципліна "Інструментальні засоби та технології програмування" є складовою циклу професійної підготовки фахівців освітнього рівня "магістр" за спеціалізацією «Теорія та технологія програмування» освітньої програми «Інформатика».

4. Завдання (навчальні цілі):

набуття знань, умінь та навичок (компетенцій) на рівні новітніх досягнень у програмуванні, відповідно до освітньої кваліфікації «Магістр з комп’ютерних наук за спеціалізацією теорія та технологія програмування». Зокрема, розвивати:

- здатність аналізувати сучасні світові тенденції розвитку комп’ютерних наук та уявляти перспективи розвитку інформаційних технологій, моделювати процеси розвитку і трансформації інформаційно-комунікаційних технологій в практичній професійній роботі;

• здатність розробляти і координувати процеси, фази та ітерації життєвого циклу програмних систем на основі застосування відповідних моделей, методів та технологій розробки програмного забезпечення.

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати суть і підґрунтя парадигми функціонального програмування, знати засоби підтримки функціонального стилю програмування та їх подання	Семінарське заняття, доповідь	Тест, 60% правильних відповідей	25%
РН1.2	Знати суть і підґрунтя парадигми логічного програмування.	Семінарське заняття, доповідь	Тест, 60% правильних відповідей	25%
РН1.3	Знати суть декларативних засобів стосовно об'єктно-орієнтованої парадигми, знати засоби декларативного програмування для .Net та Java.	Семінарське заняття, доповідь	Тест, 60% правильних відповідей	20%
РН2.1	Вміти застосовувати функціональний стиль програмування. Вміти застосовувати логічне програмування	Семінарське заняття, самостійна робота	Доповідь	15%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Семінарське заняття, самостійна робота	Доповідь	10%
РН4.1	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Самостійна робота	Доповідь	5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	РН 1.1	РН 1.2	РН 1.3	РН 2.1	РН 3.1	РН 4.1
Програмні результати навчання (з опису освітньої програми)						
ПРН18.2. Аналізувати, оцінювати і обирати методи, сучасні програмно-апаратні інструментальні та обчислювальні засоби, технології, алгоритмічні та програмні рішення для ефективного виконання конкретних виробничих задач з розробки програмного забезпечення.	+	+	+			
ПРН20.2. Мотивовано обирати технології програмування для розв'язання завдань створення і супроводження програмного забезпечення.				+	+	+

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

1. Контрольна робота (тест) 1: РН 1.1., РН 1.2 — 35 балів/35 балів.
2. Контрольна робота (тест) 2: РН1.2, РН1.3 - 35 балів/35 балів.
3. Підготовка доповіді: РН 2.1, РН 3.1, РН 4.1 – 30 балів/30 балів.

- підсумкове оцінювання у формі заліку. Виставляється за результатами роботи студентами впродовж усього семестру та не передбачає додаткових заходів оцінювання для успішних студентів.

Запитання для підготовки до заліку

1. Парадигма функціонального програмування.
2. Поліморфні типи та поліморфні функції. Параметричний поліморфізм. Виведення типів.
3. Визначення функцій рівняннями. Зіставлення зі зразком. Приклади.
4. Визначення функцій рівняннями. Рекурсивні функції. Приклади.
5. Лінійні та енергійні обчислення. Приклади.
6. Використання нескінчених списків. Списки арифметичних послідовностей.
7. Лінійні та енергійні обчислення. Використання нескінчених списків на прикладі списку простих чисел.
8. Лінійні та енергійні обчислення. Використання нескінчених списків на прикладі списку чисел Фібоначчі.
9. Скулемівські нормальні форми, множини диз'юнктив. Метод резолюцій для логіки висловлювань.
10. Уніфікація. Алгоритм уніфікації. Метод резолюцій для логіки предикатів.
11. Чистий (недетермінований) Пролог. Синтаксис. Логічна (декларативна) семантика.
12. Чистий (недетермінований) Пролог. Операційна семантика. Приклади.
13. Експертні системи та стратегії пошуку відповіді. Експертні системи із прямою стратегією.
14. Експертні системи з оберненою стратегією. Подання знань у вигляді правил Прологу.
15. Подання знань в експертних системах. Мережі знань.
16. Експертні системи з поясненнями. Питання “Навіщо?” (“*Why?*”).
17. Експертні системи з поясненнями. Питання “Як?” (“*How?*”).
18. Декларативні засоби Java-програмування на прикладі веб-служб.
19. Декларативні засоби .Net програмування на прикладі веб-служб.
20. Веб-служби (*Web Services*) та сервісно-орієнтована архітектура (COA). Стандарти веб-служб. Протокол *SOAP*. Конверт *SOAP*-повідомлення.
21. Документування веб-служб: генерація документації для сприйняття людиною (з використанням веб-браузерів), генерація документації, орієнтованої на використання програмами – *wSDL*-файли.
22. Розробка .Net веб-служб. Тест-форми веб-служб.
23. Розробка клієнтських .Net програм для веб-служб. Утиліта *WSDL.exe*.
24. Розробка веб-служб на *Java*.
25. Розробка клієнтських *Java* -програм для веб-служб.

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. *Контрольна робота 1 (тест): до 7 тижня семестру.*
2. *Контрольна робота 2 (тест): до 14 тижня семестру.*
3. *Підготовка доповіді : до 10 тижня семестру.*

Студент має право на одне перескладання контрольної роботи із можливістю отримання максимально 10 балів за кожну. Термін перескладання визначається викладачем.

У разі неякісного виконання лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторну роботу та доповідь після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

7.3 Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59
Зараховано / Passed	60-100
Не зараховано / Fail	0-59

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять

№ лекції	Назва лекції	Кількість годин		
		Лекції	Семинарські заняття	Самостійні роботи
	Частина 1. Парадигма функціонального програмування			
1	Тема 1. Основні концепції функціональної парадигми.		2	6
2	Тема 2. Функції. Типи функцій, визначення функцій рівняннями, застосування функцій. Автоматичне виведення типів. Функції від кількох аргументів. Карінг.		4	6
3	Тема 3. Визначення функцій кількома рівняннями. Рекурсивні функції та теорема про нерухому точку.		4	6
4	Тема 4. Поліморфні типи та поліморфні функції. Параметричний поліморфізм.		4	6
5	Тема 5. Лінійні та енергійні обчислення. Нескінчені списки. Списки арифметичних послідовностей.		4	6
6	Тема 6. List comprehension. Приклади.		2	6
	<i>Контрольна робота 1</i>		2	
Всього по частині 1			22	36
	Частина 2. Парадигма логічного програмування. Парадигма об'єктно-орієнтованого програмування			
7	Тема 7. "Чистий" (недетермінований) Пролог. Синтаксис, логічна семантика.		2	6
8	Тема 8. Операційна семантика Пролог-програм та метод резолюцій.		2	6
9	Тема 9. Процесори для мови Пролог. Використання відкатів.		2	6
10	Тема 10. Специфіка техніки програмування у Пролозі.		2	6
11	Тема 11. Використання Прологу при реалізації експертних систем.		2	6
12	Тема 12. Експертні системи з поясненнями. Експертні системи в умовах неповної визначеності.		2	6
13	Тема 13. "Чистий" (недетермінований) Пролог. Синтаксис, логічна семантика.		2	6
14	Тема 14. Об'єкти та метадані. Рефлексія об'єктів. Декларативний стиль програмування у Java. Анотації.		2	6
15	Тема 15. Анотації, розробка анотацій. Приклад.		2	6
16	Тема 16. Застосування декларативних засобів програмування у Java та .Net на прикладі технології веб-сервісів SOAP		2	6
	<i>Контрольна робота 2</i>		2	
Контроль за підсумками доповіді				10
Всього по частині 2			22	70
ВСЬОГО			44	106

Загальний обсяг 150 год., в тому числі:

Семинарські заняття - **44 год.**

Самостійна робота - **106 год.**

9. Рекомендовані джерела:

Основна

1. Lipovača M. Learn You a Haskell for Great Good! Miran Lipovača. : No Starch Press», 2011.
2. Братко И. Программирование на языке PROLOG для искусственного интеллекта. М.: Мир, 1990.
3. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. М.: Наука, 1983.
4. Марселлус Д. Программирование экспертных систем на Турбо-Прологе, М., Финансы и статистика, 1994. 410с.
5. Хохгуртль Б., C# и Java: межплатформенные Web-сервисы, М., Кудиц-образ, 2004, 410с.
6. Машнин Т.С. Web-сервисы Java, БХВ-Петербург, 2012, 560 с.

Додаткові:

1. Ин Ц., Соломон Д. Использование Турбо-Пролога. М.: Мир, 1990.
2. Стерлинг Л., Шапиро Э. Искусство программирования на языке Пролог. М.: Мир, 1990.
3. Стобо Дж. Язык программирования Пролог. М.: Радио и связь, 1993.
4. Клоксин У., Меллиш К. Программирование на языке пролог. М.: Мир, 1987.
5. Янсон А. Турбо-Пролог в сжатом изложении. М.: Мир, 1991.
6. Душкин Р. В. Справочник по языку Haskell. М.: ДМК Пресс, 2008.
7. Душкин Р. В. Практика работы на языке Haskell. М.: ДМК Пресс, 2009.
8. Роганова Н. А. Функциональное программирование, 2002.
9. Филд А., Харрисон П. Функциональное программирование. М.: Мир, 1993.
10. Хендерсон П. Функциональное программирование. Применение и реализация. М.: Мир, 1983.
11. Хьюдак П., Петерсон Дж., Джозеф Фасел Дж. Мягкое введение в Haskell. Учебник.