МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
імені адмірала Макарова
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
КОМП'ЮТЕРНИХ НАУК ТА УПРАВЛІННЯ ПРОЕКТАМИ

Код РПНД

ЗАТВЕРДЖЕНО
Проректор з науково-педагогічної роботи
_____ Слободян С.О.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Program of the Discipline

ДЕКЛАРАТИВНІ МОВИ ПРОГРАМУВАННЯ
Declarative programming languages

рівень вищої освіти: перший (бакалаврський)

тип дисципліни: вибіркова

мова(и) викладання: українська

Миколаїв – 2022 рік

Робоча програма навчальної дисципліни "Декларативні мови програмування"

« 26 » грудня 2022 року. –17 с.

Розробник:
В.К. Партас, доцент кафедри інформаційних управляючих систем та технологій, канд. техн. наук.

Проєкт робочої програми навчальної дисципліни "Декларативні мови програмування" розглянуто на засіданні кафедри інформаційних управляючих систем та технологій.
Протокол № 5 від « 27 » грудня 2022 р.
Завідувач кафедри_____ /І. Л. Михелєв/

Проєкт робочої програми навчальної дисципліни "Декларативні мови програмування" розглянуто Науково-методичною комісією Навчально-наукового інституту комп'ютерних наук та управління проєктами.
Протокол № 5 від « 27 » грудня 2022 р.
Голова_____ /Т. А. Фаріонова/

Проєкт робочої програми навчальної дисципліни "Декларативні мови програмування" погоджено з навчальним відділом.
Начальник відділу_____ /А. В. Лабарткава/

Робоча програма навчальної дисципліни "Декларативні мови програмування" затверджена Навчально-методичною радою НУК.
Протокол № 10 від «28» грудня 2022 р.

Голова НМР НУК_____ /Є. І. Трушляков/

© НУК, 2022

Зміст

Вступ 3

1. Опис навчальної дисципліни 5

2. Мета вивчення навчальної дисципліни 6

3. Передумови для вивчення дисципліни 6

4. Очікувані результати навчання 6

5. Програма навчальної дисципліни 7

6. Методи навчання, засоби діагностики результатів навчання та методів їх демонстрування 11

7. Форми поточного та підсумкового контролю 12

8. Критерії оцінювання результатів навчання 12

9. Засоби навчання 15

10. Рекомендована література 15

Вступ

Анотація

Зростання потужності комп’ютерів поступово впливає на технології програмування. Недоліки, які характерні для декларативних мов програмування («збірка сміття», незмінність змінних тощо), не є суттєвими. Навпаки, розвиток традиційних імперативних мов програмування здійснюється в бік впровадження характерних рис логічного, та особливо функціонального парадигм програмування. Це спонукає програміста розробляти більш надійний і короткий код, і, таким чином, скорочується час на розробку програмних засобів.

Вивчення дисципліни дає можливість ознайомитись з конкретними реалізаціями парадигм логічного і функціонального програмування. Знання арсеналу інструментальних засобів декларативного програмування допоможе у майбутнього приймати оптимальні рішення при проектуванні програмного забезпечення.

Ознайомлення з мовами декларативного програмування доцільно також дослідникам, робітникам інженерних галузей, що не пов’язані з розробкою інформаційних систем. Основна особливість декларативного підходу – програмувати в стилі опису предметної області – дає можливість основні зусилля спрямувати на рішення задачі, а не на роботу з пам’яттю комп’ютера. Таким чином, можна самостійно, без допомоги кваліфікованого програміста, отримати необхідне рішення.

Дисципліна «Декларативні мови програмування» передбачає ознайомлення з основними інструментальними засобами декількох функціональних і логічних мов програмування, розгляд особливостей реалізації рекурсивних процесів, структурування даних, методів пошуку та розв’язання складних логічних задач. Навчання складається з двох модулів відповідно до змісту навчального матеріалу.

Ключові слова: декларативне програмування, функціональне програмування, логічне програмування, рекурсивні функції, функціонали, замикання, правила, механізм повернення, відтинання

Abstract

The growth of computer power is gradually affecting programming technology. The shortcomings that are characteristic of declarative programming languages ("garbage collection", invariance of variables, etc.) are not significant. On the contrary, the development of traditional imperative programming languages is carried out towards the introduction of the characteristic features of logical and especially functional programming paradigms. This encourages the programmer to develop more reliable and shorter code, and thus reduces the time to develop software.

The study of the discipline provides an opportunity to get acquainted with specific implementations of the paradigms of logical and functional programming. Knowledge of the arsenal of declarative programming tools will help in the future to make optimal decisions when designing software.

Familiarity with declarative programming languages is also useful for researchers, engineers who are not involved in the development of information systems. The main feature of the declarative approach - to program in the style of the description of the subject area - allows you to focus on solving the problem, rather than working with computer memory. Thus, you can independently, without the help of a qualified programmer, get the necessary solution.

The discipline "Declarative programming languages" provides acquaintance with the basic tools of several functional and logical programming languages, consideration of features of implementation of recursive processes, data structuring, methods of search and the decision of difficult logical problems. Training consists of two modules according to the content of the training material.

Keywords: declarative programming, functional programming, logical programming, recursive functions, closure, predicate functions, back tracking, truncation

1. Опис навчальної дисципліни

<table>
<tr><td rowspan="2">Найменування показників</td><td rowspan="2">Галузь знань, спеціальність (освітня програма), освітній рівень</td><td colspan="2">Характеристика навчальної дисципліни</td></tr>
<tr><td>Денна форма навчання</td><td>Заочна форма навчання</td></tr>
<tr><td>Кількість кредитів – 5</td><td></td><td colspan="2">Вибіркова</td></tr>
<tr><td>Модулів – 1</td><td rowspan="3"></td><td colspan="2">Рік підготовки:</td></tr>
<tr><td>Змістових модулів –2</td><td rowspan="2">3-й</td><td rowspan="2">3-й</td></tr>
<tr><td>Електронний адрес РПНД на сайті Університету:</td><td rowspan="3"></td></tr>
<tr><td rowspan="2">Загальна кількість годин – 150</td><td colspan="2">Семестри</td></tr>
<tr><td>6-й</td><td>6-й</td></tr>
<tr><td rowspan="10">Тижневих годин для денної форми навчання:
аудиторних – 4;
самостійної роботи студента – 6.</td><td rowspan="10">Освітній рівень:
перший (бакалаврський)</td><td colspan="2">Лекції</td></tr>
<tr><td>30 год</td><td>6 год</td></tr>
<tr><td colspan="2">Практичні заняття</td></tr>
<tr><td>30 год</td><td>6 год</td></tr>
<tr><td colspan="2">Самостійна робота</td></tr>
<tr><td>90 год</td><td>138 год</td></tr>
<tr><td colspan="2">Індивідуальні завдання</td></tr>
<tr><td>–</td><td>–</td></tr>
<tr><td colspan="2">Вид контролю: залік</td></tr>
<tr><td colspan="2">Форма контролю: комбінована (письмовий контроль, тестовий контроль)</td></tr>
</table>

2. Мета вивчення навчальної дисципліни

Дисципліна доповнює обсяг знань технологій програмування парадигмами функціонального та логічного програмування, тому, отриманні знання корисні при аналізі даних, побудови складних інформаційних систем тощо.

Мета дисципліни – отримання практичних навичок застосування принципів парадигми декларативного програмування, базових знань програмування із застосуванням функціонального і логічного підходів до розробки програмного забезпечення та формування у студентів наступних **професійних компетентностей**:

- здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління;
- здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, для адекватного моделювання предметних областей і створення програмних та інформаційних систем;
- розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації

3. Передумови для вивчення дисципліни

Для вивчення дисципліни необхідно мати базові знання з курсів: "Вища математика" (функції, рекурсивні функції, множина), "Дискретна математика" (графи, алгебра логіки, відношення), «Теорія алгоритмів» (поняття алгоритму, машина Тьюрінга, рекурсія), «Алгоритмізація та програмування» (структури даних, масиви, списки).

4. Очікувані результати навчання

У результаті вивчення дисципліни студенти отримують наступні знання та вміння, що забезпечать зазначені у меті професійні компетентності:

Знання сучасних інструментальних засобів розробки програмного забезпечення, алгоритмів процедур і операцій; методів розв’язання складних задач на основі принципів декларативного програмування; організації структур даних і технологій їх обробки;. створення систем штучного інтелекту.

Вміння створювати ефективні алгоритми для обчислювальних задач інформаційних систем та систем підтримки прийняття рішень із застосуванням принципів декларативного програмування; обирати парадигму програмування для оптимального вирішення задач; проектувати та розробляти програмне забезпечення із застосуванням функціональної та логічної парадигм програмування

5. Програма навчальної дисципліни

Модуль 1

Змістовий модуль 1. Основи програмування

Тема 1 Вступ до парадигми декларативного програмування

Загальна характеристика декларативних мов програмування. Особливості використання змінних. Функції та роль функцій при композиції програм. Особливості виклику функцій. Лямбда вирази як функціональні константи. Псевдофункції.. [1] стор.11-19, 120-134, [2] стор.14-20, [5] стор.5-28, [10] стор.32-49

Тема 2 Основні конструкції мов програмування.

Механізм виведення типів. Правила побудови виразів. Співставлення зі зразком. Прості структури даних. Списки. Рекурсивна обробка списків. Особливості застосування масивів. [5] стор.30-35, [3] стор.17-30, [4] стор.75-93, [9] стор.8-20,

Тема 3 Мови логічного програмування

Предикатні функції. Факти і правила. Декларативний і процедурний сенс правил. Правила уніфікації та конкретизація змінних. Методи пошуку рішень, пряме трасування. Використання обмежень при пошуку рішень [7] стор.37-59, [6] гл. 1,2

Тема 4.Методи застосування функцій.

Підвищення рівня абстракції при розробки функцій. Функціонали. Повернення функції як результат виконання функції. Послідовності. Замикання. Відкладені обчислення. [1] стор.71-85, [4] стор.93-112, [9] стор.34-40, [3] стор.66-74

Змістовий модуль 2. Стратегії рішення задач на основі принципів декларативного програмування

Тема 5 Складні структури даних в мовах декларативного програмування

Вбудовані структури даних. Засоби побудови власних структур даних. Послідовності і «ліниві» обчислення. Генератори. Застосування замикань для побудови генераторів. Обробка даних нескінченої довжини. [3] стор.34-51,56-65, [4] стор.58-67, [5] стор.36-41, [9] стор.26-28

Тема 6 Структури даних в мовах логічного програмування.

Складні терми. Структури. Властивості уніфікації структур. Списки. Представлення та обробка списків. Рекурсивна організація даних. Дерева. Рекурсивна обробка дерев і списків. [6] гл. 3-5, [7] гл. 3, [5] гл.19

Тема 7 Стратегії вирішення задач у декларативному програмуванні

Використання методу декомпозиції для рішення складних задач. Зведення задач до підзадач. Побудова кон’юктивних правил. Пошук з перевагою. Планування виконання завдань. [6] гл. 6, 8, [7] , [5] гл.21, [8] гл. 2

Тематичний план навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	Денна форма						Заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	пр	лб	ін	с.р		л	пр	лр	ін	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Модуль 1												
Змістовий модуль 1. Основи програмування												
Тема 1. Вступ до парадигми декларативного програмування	18	4	4			10	33	2	2			10
Тема 2. Основні конструкції мов програмування	19	4	4			11						16
Тема 3 Мови логічного програмування	18	4	4			10	20	2	2			21
Тема 4 Методи застосування функцій.	20	4	4			12	22					20
Разом за змістовим модулем 1	75	16	16		-	43	75	4	4			67
Змістовий модуль 2. Стратегії рішення задач на основі принципів декларативного програмування												
Тема 5 Складні структури даних в мовах декларативного програмування	22	4	4			13	46	2	2			21
Тема 6 Структури даних в мовах логічного програмування.	26	4	6			16						24
Тема 7 Стратегії вирішення задач у декларативному програмуванні	27	6	4			18	29					26
Разом за змістовим модулем 2	75	14	14			47	75	2	2			71
Усього годин	**150**	**30**	**30**		**-**	**90**	**150**	**6**	**6**			**138**

Теми практичних занять

№ з/п	Назва теми	Кількість годин	
		Денна форма	Заочна форма
Змістовий модуль 1			
1	Функціональний підхід до вирішення задач	2	4
2	Лямбда вирази та визначення функцій	2	
3	Структури даних і рекурсивні функції	4	
4	Функції більш високого порядку	4	
5	Замикання та відкладені обчислення	4	
Змістовий модуль 2			
6	Складання простого правила	4	2
7	Управління пошуком рішень	2	
8	Рекурсивна обробка даних	4	
9	Робота з базою даних і знань	4	
	Разом	30	6

Самостійна робота

Одним з основних чинників успішного засвоєння матеріалів навчальної дисципліни є самостійна робота студентів. Основні види і розподіл часу самостійної роботи представлений в таблиці:

№ з.п.	Найменування	Кількість годин			
		Змістовий модуль 1		Змістовий модуль 2	
		Денна форма	Заочна форма	Денна форма	Заочна форма
1.	Вивчення лекційного матеріалу та відповідних розділів рекомендованої літератури.	8	14	8	14
2.	Підготовка до практичних занять і оформлення звітів з виконання кожної практичної роботи.	9	14	9	14
3.	Вивчення інструментальних засобів середовища програмування для успішного виконання практичних робіт.	10	17	12	19

№ з.п.	Найменування	Кількість годин			
		Змістовий модуль 1		Змістовий модуль 2	
		Денна форма	Заочна форма	Денна форма	Заочна форма
4.	Ознайомлення з бібліотечними модулями відповідних мов програмування при виконанні практичних робіт.	10	16	12	18
5.	Підготовка до поточного та модульного контролів з навчальної дисципліни.	6	6	6	6
Разом		**43**	**67**	**47**	**71**

6. Методи навчання, засоби діагностики результатів навчання та методів їх демонстрування

Набуття здобувачами компетенцій забезпечується через засвоєння навчального матеріалу у вигляді лекційних, практичних занять, самостійної роботи студентів та контрольних тестових робіт програмованого типу.

Основними методами навчання з дисципліни є:

1. Пояснювально-ілюстративний метод, за допомогою повідомлення та засвоєння інформації на лекційних та практичних заняттях словесними й наочними засобами.

2. Проблемний метод, коли на практичних заняттях утворюються пошукові ситуації, розвивається активність, самостійність, творчі здібності здобувачів освіти.

Як наочний матеріал на лекціях застосовуються мультимедійні слайди та лекційні демонстрації.

Для зручної та ефективної організації навчального процесу, зроблено ClassRoom з навчальної дисципліни, де розташовані завдання до практичних робіт та супроводжувальні матеріали. Доступ до класу надається на першому практичному занятті за університетською електронною поштою.

З метою роз’яснення найбільш складних питань дисципліни та підвищення якості виконання практичних завдань проводяться групові та індивідуальні консультації за розкладом кафедри.

Дистанційно заняття проводяться у ClassRoom.

Для діагностики результатів навчання з дисципліни застосовуються наступні засобі:

- звіти з виконання практичних робіт;
- працююча програма відповідно завдання з практичної роботи;
- усні відповіді на запитання викладача при захисті практичних робіт;
- поточні модульні контрольні роботи у формі тестування (тестовий контроль);

Звіт з виконання практичної роботи повинен містити:

- теоретичні відомості відповідно темі роботи;
- пояснення щодо прийняття рішень з розробки програми;
- обґрунтування використання допоміжних бібліотечних функцій,.

7. Форми поточного та підсумкового контролю

Поточний контроль складається з наступних компонентів:

захист практичних робіт, які передбачені змістовим модулем;

результати самостійної роботи;

модульний тестовий контроль за результатами вивчення тем змістових модулів.

Під час захисту практичної роботи студент повинен пояснювати прийняті рішення, а також відповідати на теоретичні питання, що стосуються до теми практичної роботи. При цьому студент повинен вільно орієнтуватись по відповідної темі, чітко знати основні визначення. Можливий письмовий контроль знань для всієї або частини групи студентів за темою практичної роботи.

По завершенні змістовного модуля здійснюється поточний модульний контроль – тестування. Тестовий контроль проводиться з метою оцінки системних знань за дисципліною в межах змістового модуля. Тест містить запитання двох типів: запитання з однозначною відповіддю, що передбачає вибір вірної відповіді з декількох можливих; запитання з багатозначною відповіддю - вибір декількох правильних відповідей серед пропонованих.

8. Критерії оцінювання результатів навчання

Змістовий модуль	**Денна та заочна форми навчання**	
	Вид роботи	**Бали**
1	Практична робота № 1	5

Змістовий модуль	Денна та заочна форми навчання	
	Вид роботи	**Бали**
	Практична робота № 2	5
	Практична робота № 3	5
	Практична робота № 4	5
	Практична робота № 5	5
	Поточний тестовий контроль	25
Разом за змістовим модулем 1		50
2	Практична робота № 6	5
	Практична робота № 7	5
	Практична робота № 8	5
	Практична робота № 9	10
	Поточний тестовий контроль	25
Разом за змістовим модулем 2		50
Всього за дисципліною		**100**

Критерії оцінювання практичних робіт

Оцінювання робіт при максимальної оцінки:		**Критерії оцінювання**
5 балів	10 балів	
5	10	Робота виконана самостійно у встановлений термін. Робота містить теоретичні відомості відповідно теми та завдання, обґрунтовані пояснення щодо прийнятих рішень, тексти розроблених програм, у висновках проведена коректна інтерпретація результатів, студент представляє працюючу програму на комп'ютері, відповідає на всі запитання, які стосуються виконаної роботи та темі роботи.
4	8	Робота виконана у встановлений термін, активно консультуючись з викладачем. Робота містить теоретичні відомості відповідно теми та завдання, пояснення щодо прийнятих рішень, тексти розроблених програм, у висновках проведена коректна інтерпретація результатів, студент

Оцінювання робіт при максимальної оцінки:		**Критерії оцінювання**
5 балів	10 балів	
		представляє на комп'ютері програму, відповідає на більшість запитань, які стосуються виконаної роботи та темі роботи.
3	6	Робота виконана з порушенням встановлених термінів. Робота містить теоретичні відомості відповідно теми та завдання, тексти розроблених програм, однак відсутні пояснення щодо прийнятих рішень, у висновках проведена коректна інтерпретація результатів, студент представляє на комп'ютері програму, при відповіді на запитання, які стосуються виконаної роботи, допускає неточності та помилки.
2	4	Робота виконана з порушенням встановлених термінів. Робота містить теоретичні відомості, які не відповідають теми та завдання; тексти розроблених програм, однак відсутні пояснення щодо прийнятих рішень; у висновках проведена коректна інтерпретація результатів; студент представляє на комп'ютері програму; при відповіді на запитання, які стосуються виконаної роботи, допускає неточності та помилки.
1	2	Робота виконана з порушенням встановлених термінів. Робота містить теоретичні відомості, які не відповідають теми та завдання; тексти розроблених програм, однак відсутні пояснення щодо прийнятих рішень; висновки не відповідають виконаної роботи; студент не відповідає на більшість запитань, які стосуються виконаної роботи, або допускає неточності та помилки.
0	0	Робота не виконувалася.

Критерії оцінювання результатів поточного модульного контролю

Тип запитання	Кількість запитань	Оцінка, бали	Разом, бали
Передбачається однозначна відповідь з декількох можливих	5	2	10
Передбачається багатозначний вибір з переліку запропонованих відповідей	5	3	15
Всього	**10**		**25**

9. Засоби навчання

Технічні засоби навчання: мультимедійний проектор, персональні комп'ютери з підключенням до мережі Інтернет.

Комп'ютерні засоби навчання: Google For Education, Visual Studio Community [17,18], SWI-Prolog [19], інструментальні засоби онлайн програмування [20-23]

10.Рекомендована література

Базова

1. Хендерсон П. Функциональное программирование. Применение и реализация.-М.:Мир, 1983.-348 с.
2. Филд А., Харрисон П. Функциональное программирование. - М.:Мир, 1993.-362 с.
3. Сошников Д.В. Функциональное программирование на F# .-М.: ДМК Пресс, 2011.-192с.:ил.
4. Смит К. Программирование на F#. – Пер. с англ. – СПб. : Символ-Плюс, 2011. – 448 с., ил.
5. Сергиевский Г.М., Волчёнков Н.Г. Функциональное и логическое программирование : учеб. пособие для студ. высш. учеб. заведений.-М.:Издательский центр «Академия», 2010, 320с.
6. Братко И.. Программирование на языке Пролог для искусственного интеллекта: // Пер. с англ./ Под редакцией В.Н.Соболева. - М.: Мир., 1990. – 560с.
7. Малпас Дж. Реляционный язик Пролог и его применение // Пер. с англ./ Под редакцией В.Н.Соболева. - М.: Наука, Гл. ред. физ.-мат. лит., 1990. – 464с.
8. Партас В.К., Гайдаєнко О.В. Методичні вказівки до виконання лабораторних робіт з курсу „Функціональне та логічне програмування". – Миколаїв: НУК , - 2009. – 72с.
9. Партас В.К., Гайдаєнко О.В. Методичні вказівки до виконання лабораторних робіт з курсу „Функціональне та логічне програмування". – Миколаїв: НУК , - 2015. – 48 с.

10. Хювёнен Э., Сеппенен И. Мир Lisp'а. В 2-х т. Т.1: Введение в язык Лисп и функциональное применение // Пер. с финск. – М.: Мир, 1990. – 447 с.

Допоміжна

11. Хювёнен Э., Сеппенен И. Мир Lisp'а. В 2-х т. Т.2: Методы и системы программирования // Пер. с финск. – М.: Мир, 1990. – 319 с.
12. Стерлинг Л., Шапиро Э. Искусство программирования на языке Пролог. / Пер. с англ. - М.: Мир, 1990. – 235с.
13. Адаменко А.Н., Кучуков А.М. Логическое программирование и Visual Prolog. – СПб.: БХВ-Петербург, 2003.-992с.

Інформаційні ресурси

14. ИНТУИТ: национальный открытый университет. Курс «Функциональное программирование». http://www.intuit.ru/studies/courses/471/327/info
15. ИНТУИТ : национальный открытый университет. Курс «Логическое программирование». http://www.intuit.ru/studies/courses/558/414/info
16. ИНТУИТ : национальный открытый университет. Курс «Основы программирования на языке Visual Prolog» http://www.intuit.ru/studies/courses/12333/1180/info
17. Каталог API (Microsoft) и справочных материалов. Интерактивный учебник по Visual F# https://msdn.microsoft.com/ru-ru/library/dd233157.aspx
18. Visual Studio Community https://visualstudio.microsoft.com/ru/vs/community/
19. Сайт інтегрованого середовища мови SWI-Prolog https://www.swi-prolog.org/
20. Інструментальні засоби онлайн програмування на мові SWI-Prolog https://swish.swi-prolog.org/
21. Use F# in the browser https://fsharp.org/use/browser/
22. Інструментальні засоби онлайн програмування на мові F# https://repl.it/languages/fsharp?v2=1
23. Інструментальні засоби онлайн програмування на мові F# https://www.jdoodle.com/compile-fsharp-online/

Розробник

к.т.н., доцент кафедри інформаційних управляючих систем та технологій /В. К. Партас/