

**КАЗАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ (ФИЛИАЛ)
АВТОНОМНОЙ НЕКОММЕРЧЕСКОЙ ОБРАЗОВАТЕЛЬНОЙ
ОРГАНИЗАЦИИ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Специальность: 10.05.01 Компьютерная безопасность

Специализация: «Разработка защищенного программного обеспечения»

Формы обучения: очная

Объем дисциплины:

в зачетных единицах: 2 з.е.

в академических часах: 72 ак.ч.

Форма промежуточной аттестации: зачет

Рабочая программа по дисциплине «Функциональное и логическое программирование» по специальности 10.05.01 Компьютерная безопасность, специализации «Разработка защищенного программного обеспечения», составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования – специалитет по специальности 10.05.01 Компьютерная безопасность, утвержденного приказом Министерства науки и высшего образования Российской Федерации от «26» ноября 2020 г. № 1459, профессионального стандарта 06.032 «Специалист по безопасности компьютерных систем и сетей», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. № 533н, профессионального стандарта 06.033 «Специалист по защите информации в автоматизированных системах», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. №525н.

Рабочая программа:

обсуждена и рекомендована к утверждению решением кафедры гуманитарных дисциплин и иностранных языков от 21 марта 2025 г., протокол № 7

одобрена Научно-методическим советом Казанского кооперативного института (филиала) от «2» апреля 2025 г., протокол № 3.

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы	4
3. Перечень планируемых результатов обучения по дисциплине	5
4. Объем дисциплины и виды учебной работы.....	6
5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий.....	7
5.1. Содержание дисциплины	7
5.2. Разделы, темы дисциплины и виды занятий	7
6. Практические занятия	7
7. Лабораторные работы	7
8. Примерная тематика курсовых работ (проектов)	8
9. Самостоятельная работа студента	8
10. Оценивание результатов обучения и уровня сформированности компетенций	11
11. Учебно-методическое обеспечение дисциплины	11
12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины.....	12
13. Материально–техническое обеспечение дисциплины	13
Обновление рабочей программы дисциплины (модуля)	14

1. Цель и задачи освоения дисциплины

Целью курса является изучение основ функционального и рекурсивно-логического программирования.

Основными задачами курса является:

- освоение парадигмы функционального программирования;
- освоение парадигмы рекурсивно-логического программирования;
- знакомство с языками программирования Clojure, Prolog, Scala;
- получение опыта командной разработки.

2. Место дисциплины в структуре образовательной программы

Дисциплина «Функциональное и логическое программирование» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» основной образовательной программы – программы специалитета по специальности 10.05.01 Компьютерная безопасность специализация «Разработка защищенного программного обеспечения».

Код компетенции	Предшествующие дисциплины (модули), практики	Изучаемые в текущем семестре дисциплины (модули), практики	Последующие дисциплины (модули), практики
ПК-3.3	Производственная практика, технологическая практика	Функциональное и логическое программирование Основы облачных технологий	Проектирование организационно-распорядительных и эксплуатационно-технических документов в системах обеспечения информационной безопасности Компьютерные преступления в компьютерных системах и сетях Производственная практика, преддипломная практика

3. Перечень планируемых результатов обучения по дисциплине

Изучение дисциплины направлено на формирование у обучающихся профессиональных компетенций.

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
ПК-3.3 Способен участвовать в проектировании программных и аппаратных средств защиты информации	ПК-3.3.1 Реализует технологии поиска и анализа следов компьютерных преступлений, правонарушений и инцидентов	Знать: основы алгоритмизации и программирования Принципы построения и виды архитектуры компьютерного программного обеспечения при разработке мобильных приложений в среде Android Studio / XCode Стандарты оформления кода для используемых языков программирования на языке Java / Kotlin / Swift Уметь: применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов при разработке мобильных приложений в среде Android Studio / XCode Владеть: проектирование программных интерфейсов при разработке мобильных приложений в среде Android Studio / XCode Проектирование структур данных в среде Android Studio / XCode Разработка технической документации на компьютерное программное обеспечение с использованием существующих стандартов при разработке мобильных приложений
	ПК-3.3.2 Применяет методы и средства прогнозирования возможных путей развития новых видов компьютерных преступлений	Знать: основы алгоритмизации и программирования Стандарты оформления кода для используемых языков программирования Java / Kotlin / Swift Типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке компьютерного программного обеспечения при разработке мобильных приложений в среде Android Studio / XCode Уметь: применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов при разработке мобильных приложений в среде Android Studio / XCode Владеть: проектирование программных интерфейсов при разработке мобильных приложений Проектирование структур данных при разработке мобильных приложений в среде Android Studio / XCode
	ПК-3.3.3 Выявление индивидуальных признаков программы, позволяющих	Знать: стандарты оформления кода для используемых языков программирования Java / Kotlin / Swift Типовые решения,

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
	впоследствии идентифицировать ее автора, а также взаимосвязи с информационным обеспечением исследуемой компьютерной системы	библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке компьютерного программного обеспечения при разработке мобильных приложений в среде Android Studio / XCode Уметь: применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов при разработке мобильных приложений в среде Android Studio / XCode Владеть: проектирование программных интерфейсов при разработке мобильных приложений в среде Android Studio / XCode Разработка, изменение архитектуры компьютерного программного обеспечения и ее согласование с системным аналитиком и архитектором программного обеспечения при разработке мобильных приложений в среде Android Studio / XCode

4. Объем дисциплины и виды учебной работы

Объем дисциплины и виды учебной работы в академических часах с выделением объема контактной работы обучающихся с преподавателем и самостоятельной работы обучающихся

очная форма обучения

Вид учебной деятельности	ак. часов	
	Всего	По семестрам
		9
1. Контактная работа обучающихся с преподавателем:	35	35
Аудиторные занятия, часов всего, в том числе:	34	34
• занятия лекционного типа	-	-
• занятия семинарского типа:	34	34
практические занятия	-	-
лабораторные занятия	34	34
в том числе занятия в форме практической подготовки	-	-
Консультации	0,5	0,5
Контактные часы на аттестацию в период экзаменационных сессий	0,5	0,5
2. Самостоятельная работа студентов всего, в том числе	37	37
- подготовка курсовой работы	-	-
- проработка учебного (теоретического) материала	15	15
- выполнение домашних заданий по практическим занятиям	15	15
- подготовка к зачету	7	7
Промежуточная аттестация: зачет	-	-

ИТОГО:	часов	72	72
общая трудоемкость	зач. ед.	2	2

5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий

5.1. Содержание дисциплины

Занятия лекционного типа: отсутствуют в учебной программе

5.2. Разделы, темы дисциплины и виды занятий

очная форма обучения

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки	самостоятельная работа	
1	Функциональное программирование	-	20/20	20	ПК-3.3, ПК-3.3.1, ПК-3.3.2, ПК-3.3.3
2	Рекурсивно-логическое программирование	-	14/14	17,8	ПК-3.3, ПК-3.3.1, ПК-3.3.2, ПК-3.3.3
	Всего	-	34/34	37	

6. Практические занятия

Практические занятия не предусмотрены.

7. Лабораторные работы

№ п/п	Наименование раздела, темы дисциплины	Содержание лабораторных работ	Объем (час.)
			Очная форма обучения
1.	Функциональное программирование	Введение в Clojure. Запуск программ. Функции в Clojure. Структуры данных в Clojure. Функции map, reduce, filter и apply. Решение задач Clojure Koans. Решение задач Clojure Koans. Разбор работы и доработка программы Mire Доработка программы Mire, работа в команде над общим проектом.	20

№ п/п	Наименование раздела, темы дисциплины	Содержание лабораторных работ	Объем (час.)
			Очная форма обучения
		Доработка программы Mire, работа в команде над общим проектом. Основы Scala. Решение задач на Scala. Решение задач на Scala. Решение задач на Scala.	
2.	Рекурсивно-логическое программирование	Введение в Prolog, факты, правила, механизм вывода. Списки в Prolog. Динамическое управление фактами. Расширение SWI-Prolog для работы с грамматиками DCG. Предикат phrase, разбор примера. Реализация клиента TCP/IP. Реализация бота Mire.	14
	Всего		34

8. Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) не предусмотрены.

9. Самостоятельная работа студента

Самостоятельная работа студента при изучении дисциплины «Функциональное и логическое программирование» направлена на:

- освоение нормативных правовых актов и учебной литературы, необходимых для освоения дисциплины;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашних заданий по практическим занятиям;
- подготовку к зачету с оценкой.

Краткие рекомендации по выполнению самостоятельной работы:

Успешное усвоение дисциплины предполагает большой, упорный, серьезный, систематический труд студентов. Важнейшая его составная часть – выполнение разных видов самостоятельной работы.

1. **Составление тематического конспекта** на основе изученной основной и дополнительной учебной литературы. В тематическом конспекте за основу берется содержание темы, вопросы для обсуждения.

Этапы работы.

1.1 Конспектирование делается только после того, как прочитан или усвоен материал для конспектирования.

1.2. Необходимо мысленно или письменно составить план конспекта. По этому плану и будет строиться конспект далее.

1.3. Составление самого конспекта. Можно сказать, что конспект – это

расширенные тезисы, дополненные рассуждениями и доказательствами, содержащимися в материалах для конспекта, а также собственными мыслями и положениями составителя конспекта.

Писать конспект рекомендуется четко и разборчиво. В конспекте можно выделять места текста в зависимости от их значимости. Для этого применяются различного размера буквы, подчеркивания, замечания на полях.

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Проект

Студенты работают в команде над сервером многопользовательской сетевой риалтаймовой игры на базе Mire. Каждому студенту дается задание доработать сервер для добавления нескольких команд, реализующих определенные аспекты игры.

Например:

Buy – показывает список предметов для покупки в магазине.

Buy item – покупает предмет item.

Студентам дается пример клиента на SWI Prolog с использованием DCG и предлагается доработать его для поддержки всех команд.

Экзамен происходит в игровой форме в виде защиты проекта. В ходе экзамена соревнуются разработанные студентами программы на Prolog на доработанном студентами сервере Mire.

Оценка отлично ставится если студент выполнил все задачи лабораторных работ, представил доработки в сервер и разработал бота на Prolog, успешно участвующего в соревнованиях.

Оценка хорошо ставится если студент выполнил все задачи лабораторных работ, представил доработки в сервер.

Студентам, которые не справились с заданиями по работе в команде, для получения оценки удовлетворительно даются индивидуальные задания на Clojure, Scala и SWI Prolog. Для получения оценки удовлетворительно (зачтено) необходимо решить самостоятельно две задачи. Примеры индивидуальных заданий:

Списки. Дан список целых чисел. Написать функцию (предикат), истинную тогда и только тогда, когда:

1. Содержит полиндром длины k.
2. Фибоначчи > 3 .
3. Арифметическая прогрессия.
4. Геометрическая прогрессия.
5. Содержит полиндром $> n$.
6. Содержит Фибоначчи $> n$.
8. Сумма четных и нечетных чисел совпадает.
9. Четные и нечетные числа чередуются.

10. Максимум встречается не более 3 раз.
11. Максимум встречается не менее 2 раз.
12. НОД всех чисел $<$ минимума в списке.
13. НОК рядом стоящих чисел $>$ максимума в списке.
14. Является записью двоичного кода дерева.
15. Состоит из пар чисел, лежащих на плоскости на одной прямой.
16. Состоит из троек чисел, лежащих в 3D-пространстве на одной прямой.
17. Неубывающая последовательность, с четным минимумом и нечетным максимумом.
18. Сумма четных чисел больше суммы нечетных чисел.
19. Возрастающая последовательность с нечетным минимумом и четным максимумом.

Графы. Дан неориентированный граф в виде списка ребер. Задается с клавиатуры числом вершин и парами соединенных ребрами вершин. Написать функцию (предикат), истинную тогда и только тогда, когда граф:

1. Содержит цикл заданной длины.
2. Содержит путь заданной длины.
3. Не содержит элементарный цикл заданной длины.
4. Не содержит элем. путь заданной длины.
5. Не содержит элем. путь, длиннее чем заданное число ребер.
6. Любую пару вершин соединяет путь длинны ≤ 5 .
7. По номеру вершины вывести номера не соединённых путем с ней.
8. По номеру вершины вывести номера вершин, которые с данной соединяет путь длинны 3.
9. Даны 2 номера вершин, определить существует ли проходящий через них элементарный цикл.
10. Даны 2 номера вершин, определить соединяет ли их путь длинны
11. Пары вершин с петлями не соединят путь длинны 3.
12. Даны 2 вершины, найти кратчайший путь обходом графа в ширину.
13. Даны 2 вершины, найти кратчайший путь обходом графа в глубину.
14. Даны 2 вершины, найти кратчайший элементарный цикл, содержащий их.
15. Даны 2 вершины, определить соединяет ли их путь из вершин с меньшими номерами.
16. Дан связный граф. Определить, содержит ли граф вершину, при удалении которой он становится несвязным. Вывести вершину.
- 16.1. Дан связный граф. Определить, содержит ли граф ребро, при удалении которого граф не содержит циклов. Вывести ребро.
17. Вывести длину максимального пути, проходящего только через вершины максимальной степени.
18. Вывести длину максимального пути, проходящего только через

вершины минимальной степени без петель.

19. Вывести длину максимального пути, проходящего только через вершины с петлями.

10. Оценивание результатов обучения и уровня сформированности компетенций

Оценивание результатов обучения по дисциплине и уровня сформированности компетенций (части компетенции) осуществляется в рамках текущего контроля успеваемости и промежуточной аттестации в соответствии с оценочными материалами.

11. Учебно-методическое обеспечение дисциплины

Основная литература:

1. Ефимова, Е.А. Основы программирования на языке Visual Prolog / Е.А. Ефимова. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. — 266 с. https://biblioclub.ru/index.php?page=book_red&id=428996&sr=1

2. Кубенский, А. А. Функциональное программирование : учебник и практикум для академического бакалавриата / А. А. Кубенский. — М. : Издательство Юрайт, 2018. — 348 с. <https://biblio-online.ru/book/658E3C89-AAD5-498B-8B34-A29E1750D810/funkcionalnoe-programmirovanie>

3. Рогозин О. В. Функциональное и рекурсивно-логическое программирование: учебно-методический комплекс. Москва: Евразийский открытый институт, 2009. 139 стр. ISBN: 978-5-374-00182-2 https://biblioclub.ru/index.php?page=book_view_red&book_id=90927

Дополнительная литература:

1 The Joy of Clojure, 2-ed. M. Fogus, C. Houser. ISBN: 9781617291418. Manning Publications Co.

2 Functional Programming in Scala. Paul Chiusano, Runar Bjarnason. Manning, September 2014. ISBN 9781617290657

3 Иван Братко. Язык PROLOG (Пролог): алгоритмы искусственного интеллекта 3-е издание. 640 стр., с ил.; ISBN 5-8459-0664-4, 0-201-40375-7; 2004, 3 кв.; Вильямс.

4 Чанышев О. Г. Программирование в логике: учебное пособие Омск: Омский государственный университет, 2004. - 32 с. ISBN: 5-7779-0510-3. Электронная библиотечная система «Университетская библиотека ONLINE». URL: <http://www.biblioclub.ru/book/83722/>

5 Головешкин В. А. , Ульянов М. В. Теория рекурсии для программистов. Учебное пособие М.: Физматлит, 2006. - 146 с. ISBN: 978-5-9221-0721-1. Электронная библиотечная система «Университетская библиотека ONLINE». URL: <http://www.biblioclub.ru/book/76680/>

Электронные библиотечные системы (ЭБС) и электронные образовательные ресурсы

1. Электронно-библиотечная система ZNANIUM <https://www.znanium.com>
2. Универсальная справочно-информационная полнотекстовая база данных периодических изданий «East View» (ИВИС) <https://dlib.eastview.com/>
3. Электронно-библиотечная система «BOOK.ru» www.book.ru
4. Электронно-библиотечная система «IPRbooks» <http://www.iprbookshop.ru/>
5. ЭБС «Национальный цифровой ресурс «Руконт» <https://lib.rucont.ru/search>
6. Образовательная платформа ЮРАЙТ <http://www.urait.ru>

Электронный каталог Университета – Автоматизированная информационная библиотечная система (АИБС «МегаПро»)

12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины

Лицензионное программное обеспечение, в том числе отечественного производства:

- DsktpEdu ALNG LicSARk OLV F1 1Y Acdmc Ent (MS Windows) (подписка на ПО).

Свободно распространяемое программное обеспечение, в том числе отечественного производства:

- 7-Zip (свободный файловый архиватор с высокой степенью сжатия данных) (отечественное ПО).

- FastStone Image Viewer (программа для просмотра изображений в Microsoft Windows (лицензия бесплатного программного обеспечения)).

- Foxit Reader (Бесплатное прикладное программное обеспечение для просмотра электронных документов в стандарте PDF (лицензия бесплатного программного обеспечения)).

- Indigo (система программ для создания и проведения компьютерного тестирования знаний).

- Google Chrome, ЯндексБраузер (браузер).

- Adobe Acrobat Reader DC (• ПО для просмотра, печати и комментирования документов в формате PDF)

- Visual Studio Code (интегрированная среда разработки программного обеспечения)

Профессиональные базы данных не используются.

Информационные справочные системы:

- СПС КонсультантПлюс. Компьютерная справочная правовая система, широко используется учеными, студентами и преподавателями (подписка на ПО)

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронно-библиотечной

13. Материально–техническое обеспечение дисциплины

Образовательный процесс обеспечен учебными аудиториями для проведения учебных занятий, а именно для проведения лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещениями для самостоятельной работы студентов.

Учебная аудитория для проведения лабораторных занятий. Аудитория информационных технологий

Рабочее место преподавателя (стол, стул), рабочие места обучающихся:

Технические средства обучения:

персональный компьютер с подключением к сети «Интернет» - 15, сетевое оборудование.

Помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду института.

Рабочие места обучающихся.

Технические средства обучения: персональный компьютер - 13.

Учебные аудитории соответствуют действующим противопожарным правилам и нормам, укомплектованы учебной мебелью.

Обновление рабочей программы дисциплины (модуля)

Наименование раздела рабочей программы, в который внесены изменения

(измененное содержание раздела)

обсуждено и рекомендовано к утверждению решением кафедры
наименование кафедры

от ____ ____ 20__ г., протокол № ____

одобрено Научно-методическим советом ____ от ____ ____ 20__ г.,
протокол № ____