МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

МИНИСТЕРСТВО НАУКИ, ВЫСШЕГО ОБРАЗОВАНИЯ И ИННОВАЦИЙ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

МОО ВО Кыргызско-Российский Славянский университет имени первого Президента Российской Федерации Б. Н. Ельцина

Высшая школа цифровых технологий

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

«Функциональное и логическое программирование»

Уровень высшего образования -магистратура

Направление подготовки: 09.04.04 «Программная инженерия»

Магистерская программа -«Основы анализа и разработки приложений с большими объемами распределенных данных»

Квалификация: магистр

Фонд оценочных средств предназначен для текущего, рубежного и промежуточного контроля результатов обучения магистрантов по дисциплине «Функциональное и логическое программирование» направления 09.04.04 «Программная инженерия».

Фонд оценочных средств рассмотрен и утвержден
на заседании Высшей школы Цифровых технологий протокол №1 от 02.09.26

Директор Высшей школы Цифровых технологий:

к.т.н., доцент Хмелева И.В.

Руководитель образовательной программы:

д.т.н., профессор Лыченко Н.М.

Исполнители (разработчики):

ст. преподаватель Беляев А.А.

СОГЛАСОВАНО:
Зам. директора по учебной работе
Политехнического института КРСУ Комарцов Н.М.

Раздел 1. Перечень компетенций с указанием этапов формирования

Компетенция / уровень	Планируемый результат обучения	Оценочные средства
ПК-4 -Знать	Основы логической и функциональной парадигм и вычислительных моделей; факты, правила, предикаты, подстановки, унификацию, рекурсию и конструкты Prolog; стратегии управления вычислениями, недетерминизм и средства функционального программирования.	Блок A: устный опрос, контрольные работы № 1–4, теоретические вопросы зачёта.
ПК-4 -Уметь	Формулировать задачи в форме фактов, правил и запросов; применять рекурсивные определения, подстановки и структуры данных; разрабатывать и модернизировать функциональные и логические компоненты.	Блок B: задания реконструктивного уровня, разбор кода; блок C: практические работы № 1–14.
ПК-4 - Владеть	Разработкой и тестированием логических программ; средствами Prolog, рекурсией, функциональными структурами данных; сравнением парадигм, управлением вычислениями и решением прикладных задач.	Блок C: реализованные, протестированные программы и отчёты; блок D: демонстрация и защита решения на зачёте.

Проверка результатов включает как воспроизведение теоретических понятий, так и самостоятельное написание, отладку, тестирование и объяснение логических и функциональных программ.

Раздел 2. Технологическая карта дисциплины

Курс 1, семестр 1; 4 ЗЕТ; 18 недель; зачёт с оценкой.

Модуль (раздел РПД)	Вид	Форма контроля	Мин., балл	Макс., балл	Нед.
1. Основы логического программирования и модели вычислений	Текущий	Защита практических работ № 1–5 (факты, правила, интерпретатор, базы данных, рекурсия, унификация).	4	8	4
	Рубежный	Контрольная работа № 1: логические модели, унификация, рекурсия, дерево вычислений.	4	8	4
2. Программирование на Prolog, метапрограммирование и недетерминизм	Текущий	Защита практических работ № 6–9: порядок правил, прикладной решатель, отсечение и	5	9	8

		отрицание, недетерминизм.			
	Рубежный	Контрольная работа № 2: анализ и модификация Prolog-программы, вычислительные стратегии.	5	9	8
3. Функциональная парадигма, языки и структуры данных	Текущий	Защита практических работ № 10–12: сравнение парадигм, функциональные определения, структуры данных.	5	8	12
	Рубежный	Контрольная работа № 3: рекурсия, функциональные структуры данных, преобразование выражений.	5	8	12
4. Функции, логические условия, итерация и сопоставление с образцом	Текущий	Защита практических работ № 13–14: функции высокого порядка, предикаты и логические условия; разбор задач на сопоставление с образцом.	6	10	16
	Рубежный	Контрольная работа № 4: условия, сопоставление с образцом, перебор с возвратом и проверка программы.	6	10	16
ВСЕГО за семестр			40	70	16
Промежуточная аттестация	Зачёт с оценкой	Теоретические вопросы, практическое задание и защита программного решения.	20	30	—
СЕМЕСТРОВЫЙ РЕЙТИНГ			60	100	—

Раздел 3. Типовые контрольные задания и иные материалы

Блок А. Теоретические вопросы и рубежные контрольные работы

Модуль 1. Контрольные вопросы

1. Как представляются факты, правила, запросы и логические переменные в логическом программировании?
2. В чём различие унификации и обычного присваивания? Приведите пример подстановки.
3. Как организованы поиск и вывод ответа в абстрактной вычислительной модели?
4. Какими правилами описать отношение в базе знаний и как выполнить запрос по нему?
5. Как работают рекурсивные определения списков и деревьев?
6. Как строится дерево вычисления запроса и как определяется успешность вывода?

Модуль 2. Контрольные вопросы

1. Как влияет порядок правил и целей на выполнение программы Prolog?
2. Для чего применяются арифметические вычисления и символьные структуры в Prolog?
3. Что такое отсечение и в каких случаях оно влияет на множество ответов?
4. Как реализуется отрицание как неуспех и каковы ограничения его применения?
5. Чем объясняется недетерминированность выполнения логической программы?
6. Как диагностировать наличие лишних ответов, зацикливания и неверного перебора?

Модуль 3. Контрольные вопросы

1. Чем функциональная парадигма отличается от императивной и логической?
2. Что такое чистая функция и почему отсутствие побочных эффектов упрощает тестирование?
3. Какие формы рекурсии используются в функциональных языках?
4. Как определяются и преобразуются списки, кортежи и иные структуры данных?
5. Какие средства предоставляет выбранный функциональный язык для обработки коллекций?
6. Какими критериями сравнивать разные реализации одной задачи?

Модуль 4. Контрольные вопросы

1. Как описать функции высокого порядка и стратегию вычисления выражений?
2. Что понимают под логическим условием, предикатом и проверкой истинности?
3. Как используются сопоставление с образцом и разбор альтернативных ветвей?
4. Чем перебор с возвратом отличается от детерминированной итерации?
5. Какие приёмы применяют для обработки неуспеха вычисления?
6. Как проверять корректность функций с несколькими образцами и пограничными случаями?

Варианты рубежных контрольных работ

Контрольная работа № 1

Задания выполняются по варианту преподавателя; требуется представить исходный код, результаты проверок и краткое обоснование решения.

1. Составить небольшую базу знаний о связях между объектами, записать не менее шести фактов и трёх правил. Привести четыре запроса с ожидаемыми результатами.

2. Для заданного рекурсивного предиката обработки списка вручную показать последовательность унификаций и дерево вывода двух запросов.

3. Разработать рекурсивный предикат поиска элемента в дереве и проверить его на пустом и непустом деревьях.

Контрольная работа № 2

Задания выполняются по варианту преподавателя; требуется представить исходный код, результаты проверок и краткое обоснование решения.

1. Для программы Prolog с двумя альтернативными правилами проследить порядок поиска и объяснить, какие ответы будут получены и в какой последовательности.

2. Разработать прикладной решатель (например, проверка допустимости расписания или отбор объектов по критериям) с использованием правил и арифметических ограничений.

3. Найти в предложенной программе ошибочное применение отсечения или отрицания, исправить код и показать тест, подтверждающий различия.

Контрольная работа № 3

Задания выполняются по варианту преподавателя; требуется представить исходный код, результаты проверок и краткое обоснование решения.

1. Реализовать обработку списка в выбранном функциональном языке двумя способами: рекурсией и через стандартные функции высшего порядка; сопоставить решения.

2. Описать алгебраический тип либо аналогичную структуру данных и реализовать функции преобразования и агрегирования её элементов.

3. Разобрать выражения с вложенными вызовами и показать последовательность вычислений, учитывая особенности выбранного языка.

Контрольная работа № 4

Задания выполняются по варианту преподавателя; требуется представить исходный код, результаты проверок и краткое обоснование решения.

1. Реализовать функцию (или систему предикатов) с несколькими случаями сопоставления с образцом, предусмотреть обработку недопустимого входа.

2. Сконструировать задачу поиска альтернативных решений; объяснить, как осуществляется перебор и когда происходит возврат.

3. Составить минимальный комплект тестов для ветвлений, граничных случаев и отсутствия подходящего решения.

Блок В. Задания реконструктивного уровня

Выполняются по исходному примеру и предполагают анализ, исправление или развитие готового решения:

- Модуль 1. Дополнить базу фактов отношениями второго уровня; изменить рекурсивные правила и объяснить изменение ответов на запросы.
- Модуль 2. Сравнить две Prolog-программы, различающиеся порядком правил; исправить неудачную стратегию перебора; обосновать применение либо отказ от отсечения.
- Модуль 3. Переписать императивную программу обработки последовательности в функциональном стиле; сохранить поведение и разработать сравнительные тесты.

- Модуль 4. Изменить определение функций и образцов так, чтобы были обработаны дополнительные варианты входа; объяснить порядок выполнения и корректность.

Блок С. Практико-ориентированные задания

Практические занятия соответствуют темам и объёму практической подготовки, указанным в РПД. Для каждого задания представляются исходные тексты программы, примеры выполнения, тесты и краткий отчёт. Язык для функциональных заданий выбирается преподавателем (например, Haskell, OCaml или Lisp).

Практическая работа № 1 (модуль 1). Основы логического программирования. Создать базу фактов и правил, выполнить запросы с переменными, объяснить полученные подстановки.

Практическая работа № 2 (модуль 1). Абстрактный интерпретатор. Реализовать и протестировать простые правила вычисления в учебном интерпретаторе.

Практическая работа № 3 (модуль 1). Базы данных и предикаты. Описать отношения через предикаты и выполнить серию запросов на выборку и соединение отношений.

Практическая работа № 4 (модуль 1). Рекурсивная обработка данных. Реализовать рекурсивную обработку списков и деревьев, проверить базовый и рекурсивный случаи.

Практическая работа № 5 (модуль 1). Унификация и дерево вывода. Трассировать унификацию; построить дерево вычисления двух запросов.

Практическая работа № 6 (модуль 2). Порядок правил и целей в Prolog. Изменить порядок правил и целей и сопоставить последовательности и множества ответов.

Практическая работа № 7 (модуль 2). Прикладной решатель на Prolog. Разработать решатель с арифметическими и символьными правилами.

Практическая работа № 8 (модуль 2). Отсечение и отрицание. Применить управление поиском и ввод-вывод, указать влияние на ответы.

Практическая работа № 9 (модуль 2). Недетерминизм. Исследовать альтернативные решения, выявить и исправить причины нецелевого перебора.

Практическая работа № 10 (модуль 3). Сравнение парадигм. Решить одну задачу императивно, логически и функционально; сравнить структуру решений.

Практическая работа № 11 (модуль 3). Типовая функциональная задача. Определить функции для преобразования последовательностей и показать тесты.

Практическая работа № 12 (модуль 3). Структуры данных. Определить и преобразовать списки, кортежи или иные структуры.

Практическая работа № 13 (модуль 4). Функции высокого порядка. Реализовать функциональную композицию/агрегацию, изучить стратегию вычислений.

Практическая работа № 14 (модуль 4). Логические условия. Разработать набор функций или предикатов для проверки сложных условий и ветвлений.

Требования к оформлению практических работ

- Формулировка задачи, входные и выходные данные, используемый язык/среда и версия инструментария.
- Исходный код и краткое пояснение ключевых конструкций, рекурсии или стратегии вычисления.

- Набор тестов: типовой случай, граничный случай, ошибочный или пустой ввод -по применимости.
- Фактические результаты тестирования, сопоставление с ожидаемыми результатами, выводы.
- Устная защита: демонстрация программы, объяснение выбранного решения и ответы на вопросы преподавателя.

Блок D. Промежуточная аттестация -зачёт с оценкой

Зачёт проводится в 1-м семестре и включает теоретическую часть, практическое программное задание и защиту представленного решения. Для промежуточной аттестации используется накопленный результат текущего контроля и отдельная оценка зачёта (до 30 баллов).

Примерные зачётные билеты

Билет № 1. Вопрос: Факты, правила и логические переменные. Практическая часть: Построить базу знаний и написать запрос с двумя переменными.

Билет № 2. Вопрос: Унификация и дерево вычисления. Практическая часть: Показать шаги унификации для рекурсивного предиката.

Билет № 3. Вопрос: Рекурсия в логическом программировании. Практическая часть: Реализовать проверку принадлежности элемента списку.

Билет № 4. Вопрос: Дерево поиска Prolog; порядок правил и целей. Практическая часть: Объяснить результаты исполнения программы при изменении порядка правил.

Билет № 5. Вопрос: Отсечение, отрицание и недетерминизм. Практическая часть: Исправить фрагмент Prolog-программы с неверным управлением поиском.

Билет № 6. Вопрос: Функциональные конструкции и рекурсия. Практическая часть: Реализовать преобразование списка рекурсией и стандартной функцией высшего порядка.

Билет № 7. Вопрос: Структуры данных в функциональном языке. Практическая часть: Определить структуру и функцию вычисления агрегата по её элементам.

Билет № 8. Вопрос: Функции высокого порядка, условия, сопоставление с образцом. Практическая часть: Обработать набор образцов и проверить крайние случаи.

Раздел 4. Методические материалы, процедуры и критерии оценивания

Текущий контроль оценивается по защищённым практическим работам и собеседованию, рубежный -по контрольным работам. Каждое подтверждение результата оценивается по содержанию, корректности кода и аргументации, а не только по факту сдачи.

Этап	Вид	Максимум	Критерии начисления баллов
Модуль 1	Текущий	8	Корректность фактов и правил -2; тестирование запросов -2; рекурсия и обработка данных -2; демонстрация и объяснение -2.
	Рубежный	8	Унификация и подстановки -3; дерево вывода -3; аргументация решения -2.
Модуль 2	Текущий	9	Прикладная программа Prolog -4; управление поиском -3; тесты и пояснение -2.

	Рубежный	9	Анализ вычисления -3; исправление ошибок и недетерминизма -4; проверка результата -2.
Модуль 3	Текущий	8	Реализация функций -3; работа со структурами -3; тесты и пояснение -2.
	Рубежный	8	Корректность алгоритма -3; функциональный стиль -3; анализ вычислений -2.
Модуль 4	Текущий	10	Функции высокого порядка -4; сложные логические условия -3; контроль крайних случаев -3.
	Рубежный	10	Сопоставление с образцом -4; перебор и возврат -3; тестирование и аргументация -3.
Зачёт с оценкой	Итоговая аттестация	30	Теоретический вопрос -10; практическое задание -14; демонстрация, объяснение и ответы на вопросы -6.

Критерии выставления баллов и итоговой оценки

- За семестр можно получить максимум 70 баллов; зачёт -максимум 30 баллов; суммарно - максимум 100 баллов.
- Пороговый минимум за работу в семестре -40 баллов, за зачёт -20 баллов, общий минимальный рейтинг -60 баллов. При невыполнении обязательной практической работы она подлежит доработке.
- 85–100 баллов -«отлично»; 70–84 -«хорошо»; 60–69 -«удовлетворительно»; 0–59 -«неудовлетворительно».
- При оценке отдельного задания: 85–100% возможных баллов -самостоятельно и корректно выполнено; 70–84% -несущественные замечания; 60–69% -базовая функциональность реализована, но имеются недочёты; менее 60% -обязательные результаты не подтверждены.
- Приоритет имеют работоспособность и проверяемость программ, соответствие заданию, корректность вычислений, аргументированность алгоритмических решений и полнота тестирования.

Процедура контроля

Преподаватель сообщает состав заданий и сроки сдачи в начале модуля. Обучающийся предъявляет код, вывод программы и отчёт, затем отвечает на вопросы по принятому решению. На рубежном контроле выполняется индивидуальный вариант. В случае доработки фиксируются замечания и повторно оценивается исправленная версия согласно действующему порядку университета.

Раздел 5. Методические указания для обучающихся

5.1. Подготовка к текущему и рубежному контролю

- По модулю 1 изучить предикаты, подстановки, унификацию, простые и рекурсивные правила; потренироваться объяснять дерево вывода.

- По модулю 2 освоить запуск запросов в Prolog, порядок правил, поиск с возвратом, отсечение, отрицание и разбор недетерминированных вычислений.
- По модулю 3 изучить базовые функциональные конструкции, рекурсию и стандартные операции со структурами данных.
- По модулю 4 отработать функции высокого порядка, логические условия, сопоставление с образцом и проверку всех ветвей функции.
- Для каждой программы фиксировать условия запуска, исходные данные, ожидаемый и фактический результат; хранить разные версии кода и материалы тестирования.

5.2. Подготовка к зачёту с оценкой

Повторить определения из блока А, выполненные практические работы и контрольные задания; подготовиться к демонстрации одного логического и одного функционального решения. На зачёте необходимо не только запустить программу, но и объяснить модель вычисления, выбор рекурсии или образцов, результаты тестов и возможные ограничения.

5.3. Обеспечение академической добросовестности

Допускается использование учебной литературы, официальной документации языков и справочных материалов с указанием источника. Обучающийся должен самостоятельно объяснять представленный код, демонстрировать выполнение и анализировать его корректность. Совпадение решений без подтверждения самостоятельного понимания не заменяет защиту работы.