

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Косенок Сергей Михайлович
Должность: ректор
Дата подписания: 11.06.2026 10:46:47
Уникальный программный ключ:
e3a68f3eaa1e62674b54f4998099d3d6bfdcf836

Оценочные материалы для текущего контроля и промежуточной аттестации по дисциплине

Функциональное программирование

Код, направление подготовки	09.03.04 Программная инженерия
Направленность (профиль)	Программное обеспечение компьютерных систем
Форма обучения	заочная
Кафедра-разработчик	автоматики и компьютерных систем
Выпускающая кафедра	автоматики и компьютерных систем

Типовые контрольные задания

Текущий контроль в форме контрольной работы проводится в виде теста с открытой формой ответов. Вопросы формулируются в виде:

– дать пояснение синтаксису (семантике) некоторого фрагмента программного кода или провести сравнительный анализ синтаксиса (семантики) нескольких вариантов программного кода;

– построить функцию, реализующую математическое выражение (указано некоторое математическое выражение).

– определена функция (задано определение функции на языке Common Lisp), необходимо определить, какое значение вернет вызов (задан вызов функции и некоторый определенный набор фактических аргументов);

– реализовать рекурсивную функцию (не используя специальные функции и функционалы), обрабатывающую список заданным образом;

– построить функционал, получающий заданное количество функциональных объектов и / или возвращающий заданную функцию.

Этап: проведение промежуточной аттестации (экзамен)

Примерный перечень вопросов:

1. Декларативное и императивное программирование. Языки функционального программирования.
2. Лямбда-исчисление Черча.
3. Основные элементы семантики и синтаксиса языка Lisp. Области применения функционального программирования.
4. Структурные единицы Lisp. Понятия атомов, списков и s-выражений.
5. Базовые функции языка Lisp. Основные функции обработки списков.
6. Точечная пара. Предикаты. Дополнительные функции обработки списков.
7. Арифметические функции. Логические функции.
8. Понятие символа. Функции связывания символа и его значения.
9. Анонимные функции, λ -выражения и λ -вызовы. Именованные функции (функции пользователя). Ключевые слова. Функциональная блокировка.
10. Организация ввода-вывода средствами языка Lisp. Функции ввода-вывода. Форматированный ввод-вывод.
11. Управляющие конструкции: последовательные вычисления. Разветвление вычислений.
12. Управляющие конструкции: условные предложения. Циклические вычисления.
13. Функционалы. Отображающие и применяющие функционалы.
14. Функции высоких порядков.
15. Энергичные и ленивые вычисления.
16. Определение рекурсии. Трассировка. CAR- и CDR-рекурсия.
17. Множественные терминальные ветви. Множественные рекурсивные ветви.
18. Представление списков через списочные ячейки и через точечные пары. Сборка мусора.

Примерные практические задания промежуточной аттестации (экзамен) формируются следующим образом.

Формулировка задания: Реализовать функцию, выполняющую ... (обработку списка, поиск значения, сортировку и др.). Исходные данные ... (перечисление величин или данных) передаются в качестве параметров, функция возвращает ... (указание возвращаемого значения или результата выполнения функции). Привести пример использования этой функции.

Реализуемая функция направлена на решение следующих задач (по отдельности или в сочетаниях):

- организация обработки линейного (вложенного) списка;
- построение рекурсивной функции;
- вычисление математических зависимостей;

- построение вычислительных алгоритмов;
- построение функционалов, обобщающих решение класса типовых задач;
- использование анонимных функций и/или функциональных объектов.

При выполнении задания студент демонстрирует знание синтаксиса и семантики описаний, операций и операторов, знание форматов представления данных, умения и навыки анализа и формализации задачи, работы со справочными и нормативными материалами, выбора типов данных, операций и конструкций языка Common Lisp.

Примерный перечень практических заданий:

- Реализовать функцию, осуществляющую подсчет количества четных чисел в многоуровневом списке.
- Реализовать функцию, осуществляющую подсчет количества списков в многоуровневом списке.
- Реализовать функцию, осуществляющую подсчет сумму чисел в многоуровневом списке.
- Реализовать функцию, осуществляющую подсчет сумму нечетных чисел в многоуровневом списке.
- Реализовать функцию, осуществляющую подсчет количества пустых списков в многоуровневом списке.
- Реализовать функцию, осуществляющую подсчет количества отрицательных чисел в многоуровневом списке.
- Реализовать рекурсивную функцию (не используя специальные функции и функционалы), получающую список (четной длины) чисел и возвращающую список, содержащий элементы, пары которых переставлены.
- Реализовать рекурсивную функцию (не используя специальные функции и функционалы), получающую список (четной длины) чисел и возвращающую список пар исходных элементов.
- Реализовать рекурсивную функцию (не используя специальные функции и функционалы), получающую список пар чисел и возвращающую одноуровневый список исходных чисел.