

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

Тульский филиал Финуниверситета

Кафедра «Математика и информатика»

СОГЛАСОВАНО

ЗАО «ЛИМ»

Генеральный директор

В.В. Куликов

«19» декабря 2024 г.



УТВЕРЖДАЮ

Директор филиала

Г.В. Кузнецов

«24» декабря 2024 г.



Н.О. Козлова, Е.В. Манохин

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ
ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ**

для студентов, обучающихся по направлению подготовки
09.03.02 Информационные системы и технологии,
образовательная программа «Информационные технологии в экономике»,
профиль «Информационные технологии в управлении цифровой
экономикой»

Рекомендовано Ученым советом Тульского филиала Финуниверситета
(протокол от 24 декабря 2024 г. № 22)

Одобрено кафедрой «Математика и информатика»
(протокол от 02 декабря 2024 г. № 5)

Тула 2024

«Функциональное и логическое программирование»

КОД И НАИМЕНОВАНИЕ ОПОП:

09.03.02 Информационные системы и технологии

Образовательная программа - Информационные технологии в экономике

Профиль «Информационные технологии в управлении цифровой экономикой»

Общее количество заданий по дисциплине

Код компетенции	Наименование компетенции	Количество заданий
ПКП-5	Способен разрабатывать программные продукты в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению программных систем	40
Всего:		40

Количество заданий по дисциплине в одном варианте

Код компетенции	Наименование компетенции	Количество заданий
ПКП-5	Способен разрабатывать программные продукты в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению программных систем	20
Всего:		20

Типы заданий

Наименование типа заданий	Номер типа заданий
Задание комбинированного типа с выбором одного верного ответа из предложенных	1
Задание открытого типа	2
Задание открытого типа с развернутым ответом	3

Распределение заданий по компетенциям

Код компетенции	Наименование компетенции	Наименование индикаторов сформированности компетенции	Семестр	Номер задания
ПКП-5	Способен разрабатывать программные продукты в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению программных систем	1. Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом.	5	1.1, 1.2, 2.1, 2.2, 3.1, 3.2, 4.1, 4.2, 5.1, 5.2, 6.1, 6.2, 7.1, 7.2, 8.1, 8.2, 9.1, 9.2, 10.1, 10.2
		2. Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	5	11.1, 11.2, 12.1, 12.2, 13.1, 13.2, 14.1, 14.2, 15.1, 15.2, 16.1, 16.2, 17.1, 17.2, 18.1, 18.2, 19.1, 19.2, 20.1, 20.2

Типы, уровень сложности и время выполнения заданий

Код компетенции	Индикатор сформированности компетенции	Номер задания	Тип задания	Уровень сложности задания	Время выполнения задания (мин.)
ПКП-5	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	1.1, 1.2.	1	базовый	2
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	2.1, 2.2	1	базовый	2
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	3.1, 3.2	1	базовый	2
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	4.1, 4.2	1	базовый	2
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	5.1, 5.2	1	базовый	2
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	6.1, 6.2.	2	повышенный	4
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	7.1, 7.2	2	повышенный	4
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	8.1, 8.2	2	повышенный	4
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	9.1, 9.2	3	высокий	8
	Разрабатывает архитектуру и дизайн программной системы на основе анализа требований к программной системе в целом	10.1, 10.2	3	высокий	8
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	11.1, 11.2.	1	базовый	2
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	12.1, 12.2	1	базовый	2
	Проектирует пользовательское взаимодействие с программной	13.1, 13.2	1	базовый	2

	системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.				
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	14.1, 14.2	1	базовый	2
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	15.1, 15.2	1	базовый	2
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	16.1, 16.2.	2	повышенный	4
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	17.1, 17.2	2	повышенный	4
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	18.1, 18.2	3	высокий	8
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	19.1, 19.2	3	высокий	8
	Проектирует пользовательское взаимодействие с программной системой, включая пользовательский интерфейс, на основе аналитики пользовательского опыта.	20.1, 20.2	3	высокий	8

Задания, позволяющие осуществлять оценку компетенций, установленных для дисциплины

Вариант 1

Задание 1.1.

Что такое функциональное программирование?

- А) Программирование, основанное на изменении состояния объектов.
- Б) Программирование, в котором основное внимание уделяется выполнению инструкций.
- В) Программирование, основанное на применении функций к аргументам и избегании побочных эффектов.
- Г) Программирование, сосредоточенное на манипуляциях с памятью.

Ответ:

Задание 2.1.

Что такое лямбда-функция в функциональном программировании?

- А) Функция, которая объявляется заранее и используется многократно.
- Б) Анонимная функция, которая создается и используется непосредственно в месте вызова.
- В) Функция, которая принимает несколько аргументов и возвращает один результат.
- Г) Функция, которая изменяет глобальные переменные.

Ответ:

Задание 3.1.

Какое основное различие между функциональным и логическим программированием?

- А) Функциональное программирование основывается на выполнении инструкций, а логическое - на манипуляциях с памятью.
- Б) В функциональном программировании используется рекурсия, а в логическом - нет.
- В) В функциональном программировании акцент на вычисления через функции, в логическом - на решении логических задач с фактами и правилами.
- Г) В логическом программировании используются чистые функции, а в функциональном - нет.

Ответ:

Задание 4.1.

Что такое монада в функциональном программировании?

- А) Способ обработки исключений.
- Б) Конструкция, представляющая способ управления последовательными вычислениями и обертывания значений.
- В) Метод, используемый для управления памятью.
- Г) Тип функции, принимающей несколько аргументов.

Ответ:

Задание 5.1.

Что такое дедукция в логическом программировании?

- А) Процесс выполнения функций.
- Б) Процесс выполнения инструкций.
- В) Процесс вывода новых знаний на основе фактов и правил.
- Г) Процесс обработки ошибок.

Ответ:

Задание 6.1.

Парадигма программирования, основанная на логике предикатов, которая использует правила и факты для описания и решения задач – это ...

Ответ:

Задание 7.1.

Приведите пример функционального языка программирования.

Ответ:

Задание 8.1.

Конструкция, которая позволяет связывать вычисления и управлять их порядком выполнения – это ...

Ответ:

Задание 9.1.

В чём разница между функциональным и логическим программированием?

Ответ:

Задание 10.1.

Что такое ленивая оценка в функциональном программировании?

Ответ:

Задание 11.1.

Формализм исчисления предикатов первого порядка используется в парадигме программирования:

- А) процедурной
- Б) функциональной
- В) объектно-ориентированной
- Г) логической

Ответ:

Задание 12.1.

В функциональном программировании функции сами по себе могут быть переданы другим функциям в качестве аргумента или возвращены в качестве результата. Это называется ...

- А). аппликациями
- Б) функциями высших порядков
- В) рекурсивными функциями
- Г) замещениями

Ответ:

Задание 13.1.

Есть ли возможность во время исполнения программы на языках Lisp или Haskell создавать новые функции?

- А) Нет, такой возможности нет
- Б) В исключительных случаях - возможно
- В) Да, есть, и это по сути – адаптация и обучение и на Лиспе, и на Хаскеле
- Г) На Лиспе – возможно, на Хаскеле - нет

Ответ:

Задание 14.1.

В комбинаторной логике Хаскелла Карри существует лишь одна операция - ...

- А) умножение
- Б) аппликация
- В) сопоставление
- Г) замещение

Ответ:

Задание 15.1.

Функциональная программа состоит из ...

- А) Множества предикатов
- Б) Из определений различных объектов
- В) Одной функции
- Г) Отдельных определений функций

Ответ:

Задание 16.1.

Анонимная функция, которую можно создать и использовать без предварительного объявления, часто используемая для кратковременных операций – это ...

Ответ:

Задание 17.1.

Как в функциональном программировании называется монада, которая представляет собой конструкцию для работы с последовательностями элементов и которая позволяет выполнять операции над списками.

Ответ:

Задание 18.1.

Перечислите основные принципы функционального программирования.

Ответ:

Задание 19.1.

Что такое сопоставление с образцом в функциональном программировании?

Ответ:

Задание 20.1.

Что такое монада State в функциональном программировании?

Ответ:

Вариант 2

Задание 1.2.

Какой принцип лежит в основе чистых функций в функциональном программировании?

- А) Функция изменяет глобальные переменные.
- Б) Функция может возвращать разные значения при одинаковых входных данных.

В) Функция всегда возвращает одно и то же значение для одинаковых входных данных и не имеет побочных эффектов.

Г) Функция зависит от состояния системы.

Ответ:

Задание 2.2.

Что такое карринг (currying) в функциональном программировании?

А) Преобразование функции, принимающей несколько аргументов, в одну функцию, принимающую все аргументы одновременно.

Б) Преобразование функции в последовательность функций, каждая из которых принимает один аргумент. +

В) Преобразование функции в тип данных.

Г) Преобразование функции в объект.

Ответ:

Задание 3.2.

Что такое рекурсия в контексте функционального программирования?

А) Метод, при котором функция вызывает другую функцию для решения задачи.

Б) Метод, при котором функция вызывает саму себя для решения задачи.

В) Метод, при котором функция изменяет глобальные переменные.

Г) Метод, при котором функция работает с объектами.

Ответ:

Задание 4.2.

Какой язык программирования считается основным для логического программирования?

А) Java

Б) Python

В) Prolog

Г) C++

Ответ:

Задание 5.2.

Что такое «чистая» функция в функциональном программировании?

А) Функция, которая изменяет состояние системы.

Б) Функция, которая может возвращать разные значения для одинаковых входных данных.

В) Функция, которая всегда возвращает одно и то же значение для одинаковых входных данных и не имеет побочных эффектов.

Г) Функция, которая вызывает другие функции.

Ответ:

Задание 6.2.

Приведите пример логического языка программирования.

Ответ:

Задание 7.2.

Процесс преобразования функции с несколькими аргументами в функцию с одним аргументом и создания новой функции для каждого аргумента – это ...

Ответ:

Задание 8.2.

Как называется программирование, основанное на использовании монад для управления вычислениями и упорядочивания их выполнения?

Ответ:

Задание 9.2.

Каковы основные принципы логического программирования?

Ответ:

Задание 10.2.

Что такое монада Maybe в функциональном программировании?

Ответ:

Задание 11.2.

В основу какой парадигмы программирования положена модель - лямбда-исчисления Черча?

- А) императивной
- Б) продукционной
- В) аппликативной
- Г) Объектно-ориентированной

Ответ:

Задание 12.2.

Название "аппликативный стиль программирования" соответствует ...

- А) Логическому программированию
- Б) Функциональному программированию
- В) Объектно-ориентированному программированию
- Г) Процедурному программированию

Ответ:

Задание 13.2.

Термин "парадигма" в программировании означает...

- А) Язык программирования
- Б) Универсальный подход к решению проблем программирования
- В) Модель вычислений как способ структурирования информации, организации вычислений и данных
- Г) описание логики программирования

Ответ:

Задание 14.2.

Функция `setq` связывает имя, не вычисляя его. А каков побочный эффект у функции `setq`?

- А) Нет побочного эффекта
- Б) Выводит на экран само имя, а не его значение
- В) Создает указатель, то есть образует связь между символом и его значением
- Г) Выводит на экран и имя, и значение

Ответ:

Задание 15.2.

Укажите особенность логических языков программирования:

- А) программа задаёт множество возможных переходов в пространстве поиска
- Б) выполнение операторов изменяет состояние памяти
- В) применение функции к аргументам изменяет данные

Ответ:

Задание 16.2.

Какая монада в функциональном программировании представляет собой конструкцию для работы с контекстом, который может быть добавлен к результату функции?

Ответ:

Задание 17.2.

Метод, при котором функция вызывает саму себя для решения задачи – это ...

Ответ:

Задание 18.2.

Что такое дедукция в логическом программировании?

Ответ:

Задание 19.2.

Что такое «чистая» функция в функциональном программировании, и почему она важна?

Ответ:

Задание 20.2.

Что такое монада `Cont` в функциональном программировании?

Ответ:

Ключи к оцениванию заданий

Вариант 1

№ задания	Верный ответ	Критерии оценивания
1.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
2.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
3.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
4.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
5.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
6.1	Логическое программирование	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
7.1	Пример функционального языка программирования - это язык Haskell.	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
8.1	Монада	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
9.1	Функциональное программирование фокусируется на обработке данных с помощью функций, в то время как логическое программирование основано на логике предикатов и правилах для решения задач.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
10.1	Ленивая оценка в функциональном программировании означает, что вычисление функции происходит только при необходимости, что может привести к более эффективному использованию ресурсов.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
11.1	Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
12.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
13.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
14.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
15.1	Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
16.1	Лямбда-функция	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
17.1	Монада List	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
18.1	Основные принципы функционального программирования включают использование функций без побочных эффектов, применение чистых функций и применение рекурсии.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
19.1	Сопоставление с образцом в функциональном программировании — это	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования

	механизм для проверки соответствия значения определённому шаблону и выполнения соответствующих действий.	0 баллов – неверный ответ или его отсутствие
20.1	Монада State в функциональном программировании представляет собой конструкцию для работы с изменяемым состоянием, которое может быть обновлено внутри функции.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие

Вариант 2

№ задания	Верный ответ	Критерии оценивания
1.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
2.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
3.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
4.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
5.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
6.2	Пример логического языка программирования - это язык Prolog	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
7.2	Каррирование	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
8.2	Монадическое программирование	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
9.2	Основные принципы логического программирования включают использование логики предикатов, применение резолюции и использование декларативного подхода к программированию.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
10.2	Монада Maybe в функциональном программировании представляет собой конструкцию для работы с неопределёнными значениями, такими как отсутствие результата или ошибки.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
11.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
12.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
13.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
14.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
15.2	А	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
16.2	Монада Writer	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
17.2	Рекурсия	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
18.2	Дедукция — это процесс вывода новых знаний на основе имеющихся фактов и правил в логической системе. Это основной механизм работы логических программ и систем вывода.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
19.2	Чистая» функция — это функция, которая для одного и того же набора входных данных всегда возвращает один и тот же результат и не имеет побочных эффектов. Такие функции важны, так как они	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие

	упрощают тестирование, отладку и понимание кода.	
20.2	Монада Cont в функциональном программировании представляет собой конструкцию для работы с параллельными вычислениями, которая позволяет выполнять операции над функциями.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие