Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования «Новосибирский национальный исследовательский государственный университет» (Новосибирский государственный университет, НГУ)

Высший колледж информатики

Согласовано

Директор ВКИ НГУ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Г. Окунев

*подпись*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

***Декларативное программирование***

направление подготовки: *15.03.06 Мехатроника и робототехника*

направленность (профиль): *Мехатроника и робототехника*

Форма обучения: очная

Разработчики:

Власов В.Н., к.ф.-м.н., доцент кафедры Систем информатики ФИТ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель программы:

д.т. н., Назаров А.Д. . \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Новосибирск, 2020

**Содержание**

[1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы 3](#_Toc58330770)

[2. Место дисциплины в структуре образовательной программы 3](#_Toc58330771)

[3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося 3](#_Toc58330772)

[4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий 4](#_Toc58330773)

[5. Перечень учебной литературы 8](#_Toc58330774)

[6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины 8](#_Toc58330775)

[7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине 9](#_Toc58330776)

[8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине 9](#_Toc58330777)

[9. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине 9](#_Toc58330778)

Приложение 1 Аннотация по дисциплине

Приложение 2 Оценочные средства по дисциплине

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

| Результаты освоения образовательной программы  (компетенции) | В результате изучения дисциплины обучающиеся должны: | | |
| --- | --- | --- | --- |
| знать | уметь | владеть |
| ОПК-3 владение современными информационными технологиями, готовность применять современные средства автоматизированного проектирования и машинной графики при проектировании систем и их отдельных модулей, а также для подготовки конструкторско-технологической документации, соблюдать основные требования информационной безопасности | методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области проектировании систем и их отдельных модулей  *- знать принципы и концепции декларативного программирования в части как функционального, так и логического подхода, а также основные структуры и методы, связанные с декларативными языками.* | применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области проектировании систем и их отдельных модулей  *- уметь объяснить характер и порядок вычислений декларативной программы, следуя семантике и прагматике соответствующего языка и виртуальной машины, на которой он реализован.* | навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач в области проектировании систем и их отдельных модулей *- знать и понимать связи между базисными математическими понятиями (лямбда выражения, типы, классы) и конструкциями языков программирования;*  *- уметь строить иерархические конструкции в языках функционального и логического программирования.* |

# 2. Место дисциплины в структуре образовательной программы

Дисциплина *Декларативное программирование* является базовой для освоения дисциплин Модели вычислений, Объектно-ориентированное программирование.

Цель преподавания дисциплины состоит в содействии формированию способности использовать современные информационные технологии и программные средства для решения профессиональных задач.

# 3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

Трудоемкость дисциплины – 7 з.е. (252 ч)

Форма промежуточной аттестации: 1 семестр – дифференцированный зачет,

2 семестр – экзамен

|  |  |  |  |
| --- | --- | --- | --- |
| № | Вид деятельности | Семестр | |
| 1 | 2 |
| 1 | Лекции, ч | 32 | 32 |
| 2 | Практические занятия, ч | 32 | 32 |
| 3 | Лабораторные работы , ч |  |  |
| 4 | Занятия в контактной форме, ч  из них | 66 | 68 |
| 5 | из них аудиторных занятий, ч | 64 | 64 |
| 6 | в электронной форме, ч | - | - |
| 7 | консультаций, час. | - | 2 |
| 8 | промежуточная аттестация, ч | 2 | 2 |
| 9 | Самостоятельная работа, час. | 42 | 76 |
| 10 | Всего, ч | 108 | 144 |

# 4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

***1 семестр***

**Лекции (32 ч)**

|  |  |
| --- | --- |
| Наименование темы и их содержание | Объем,  час |
| Лекция 1. История Функционального программирования и его основные представители. | 2 |
| Лекция 2. Парадигмы программирования. | 2 |
| Лекция 3. Основы языка Haskell, определения функций, сопоставление с образцом (pattern matching), ветвления. Параметрический полиморфизм | 2 |
| Лекция 4. Карринг и лямбда-абстракция, замыкания (обзор реализации в др. языках программирования), Операторы композиции функции и применения | 2 |
| Лекция 5. Система типов в Haskell, кортежи, списки, множества. Базовые функции для работы с ними | 2 |
| Лекция 6. Свертки, их варианты (foldl, foldr) и их использование для обработки списков и обобщенных структур. | 2 |
| Лекция 7. Система типов в Haskell и др. языках программирования. Виды типизации: полная, сильная, статическая, явная | 2 |
| Лекция 8. Классы типов в Haskell, абстрактные типы, ad-hoc-полиморфизм. Предопределенные классы типов в Haskell | 2 |
| Лекция 9. Структуры данных в Haskell. Тип данных «Maybe», «Either» и их возможности. Рекурсивные типы – списки, деревья. Массивы в функциональном программировании | 2 |
| Лекция 10. Представление текста в Haskell и его обработка. Регулярные выражения в Haskell | 2 |
| Лекция 11. Структура программ, точка входа. Стандартный ввод-вывод. Проекты, запуск и компиляция. Обработка текстовых файлов. Общее представление о работе с бинарными файлами | 2 |
| Лекция 12. Императивное программирование на Haskell, его эмуляция. Изменяемые переменные и изменяемые состояния | 2 |
| Лекция 13. Моноиды, в математике и в практике. Функторы и аппликативные функторы | 2 |
| Лекция 14. Монады, монадные значения и монадная функция. Класс «Monad». Законы монад | 2 |
| Лекция 15. Конкретные монады и их приложения (Maybe, Either, Writer, Монады State и Cont) | 2 |
| Лекция 16. Трансформеры, монадные утилиты, MonadPlus | 2 |
| **Итого:** | **32** |

**Практические занятия (32 ч.)**

|  |  |
| --- | --- |
| Содержание практического занятия | Объем, час |
| Тема 1. Занятия 1-4  Основы языка Haskell, типизация данных и функций, возможности ghci  Сопоставление с образцом (pattern matching).  Создание простых вычислительных функций  Решение задач и обсуждение вопросов, связанных с теорией. | 8 |
| Тема 2. Занятия 5-10 Рекурсия и ко-рекурсия, Хвостовая рекурсия, представление циклов в рекурсивном виде.  Обобщенные функции и параметрический полиморфизм.  Решение задач и обсуждение вопросов, связанных с теорией. | 12 |
| Тема 3. Занятия 11-16  Обработка списков, свёртки, множества.  Бесконечные структуры данных.  Решение задач и обсуждение вопросов, связанных с теорией. | 12 |
| **Итого:** | **32** |

**Самостоятельная работа студентов (42 ч)**

|  |  |
| --- | --- |
| Перечень занятий на СРС | Объем, час |
| Изучение разделов дисциплины по учебной литературе, в том числе вопросов, не освещаемых на лекциях. Изучение предлагаемых теоретических разделов в соответствии с настоящей Программой. Учебно-методические материалы по дисциплине «Декларативное программирование» выложены на странице курса в сети Интернет | 10 |
| Подготовка к практическим занятиям, к текущему контролю знаний и промежуточной аттестации. Разбор решенных задач, самостоятельное решение задач. | 20 |
| Подготовка к дифференцированному зачету. Повторение теоретического материала по вопросам, совпадающим с темами лекций | 12 |
| **Итого:** | **42** |

***2 семестр***

**Лекции (32 ч)**

|  |  |
| --- | --- |
| Наименование темы и их содержание | Объем,  час |
| Лекция 1. Графы, деревья, обходы деревьев. Конечные автоматы в функциональном языке, связь с регулярными выражениями | 2 |
| Лекция 2. Грамматики, дерево синтаксического разбора, парсеры. Схема восходящего парсера. Парсинг сбалансированных скобок и булевых выражений. | 2 |
| Лекция 3. Шаблоны (Template) в Haskell как возможности метапрограммирования | 2 |
| Лекция 4. Расширения системы типов Haskell. Многопараметрические классы типов и функциональные зависимости. Семейства типов. GADT | 2 |
| Лекция 5. Вывод типов, модель типизации Хиндли-Милнера. Дальнейшее развитие возможностей системы типов: Idris, Agda, Clean, Curry | 2 |
| Лекция 6. Линзы и варианты ООП в Haskell | 2 |
| Лекция 7. Параллельное вычисление чисто функциональных программ. Конкурентность и связанные с этим проблемы. | 2 |
| Лекция 8. Декларативное программирование вне функциональных языков программирования. Реляционные базы данных. SQL – декларативный язык управления и описания баз. История разработки. | 2 |
| Лекция 9. Основы языка SQL: оператор SELECT. Кортежи и типы. Типы в языке SQL. Унарные реляционные операции: проекция, выборка. Основные предикаты над числами и строками. Отношения, как множества. Операции над множествами. Уникальная выборка (DISTINCT), сортировка (ORDER BY), окна, агрегирующие операции. | 2 |
| Лекция 10. Основы реляционной алгебры. Переименование. Декартово произведение отношений. Естественное соединение, θ-соединение. Внутреннее соединение. Внешнее соединение. Значение NULL. Практика: Выборка из нескольких таблиц. Внутреннее и внешнее соединение. | 2 |
| Лекция 11. Агрегация и группировка (GROUP BY). Агрегирующие операции и условия при группировке (HAVING). Суперключ. Возможный, первичный, внешний ключи. | 2 |
| Лекция 12. Понятие о нормализации отношений. Нормальные формы (1-5). Функциональные, множественные зависимости, зависимости соединения. Элементы проектирования баз данных. ER-модель. Множественность связей, 1-1, 1-n, n-n. Связи типа 0..1-n, внешнее соединение. Представление деревьев и графов. | 2 |
| Лекция 13. Вложенные запросы. Каскадная агрегация. Эквивалентные запросы и переписывание запроса. | 2 |
| Лекции 14-15. Эффективность запросов. Основные алгоритмы вычисления соединений. Понятие индекса. Оптимизация плана запроса: алгебраические свойства реляционных операций, отложенные вычисления. | 4 |
| Лекция 16. Обзор проблематики баз данных. Эффективность. Альтернативные модели данных. Свойства ACID. | 4 |
| **Итого:** | **32** |

**Практические занятия (32 ч.)**

|  |  |
| --- | --- |
| Содержание практического занятия | Объем, час |
| Тема 1. Занятия 1-4 Типы и классы типов.  Обработка текстовых файлов  Решение задач и обсуждение вопросов, связанных с теорией. | 8 |
| Тема 2. Занятия 5-8 Монады и функторы, использование возможностей утилит.  Решение задач и обсуждение вопросов, связанных с теорией. | 8 |
| Тема 3 Занятия 9-14 Реализация запросов с применением основных элементов SQL: проекция, селекция, соединения, группировка и агрегация.  Вложенные запросы. | 8 |
| Тема 4 Занятия 15-16 Эффективность запросов. Переписывание запросов. Реализация планов запросов на структурах данных в Haskell. | 8 |
| **Итого:** | **32** |

**Самостоятельная работа студентов (76 ч)**

|  |  |
| --- | --- |
| Перечень занятий на СРС | Объем, час |
| Изучение разделов дисциплины по учебной литературе, в том числе вопросов, не освещаемых на лекциях. Изучение предлагаемых теоретических разделов в соответствии с настоящей Программой. Учебно-методические материалы по дисциплине «Декларативное программирование» выложены на странице курса в сети Интернет | 10 |
| Подготовка к практическим занятиям, к текущему контролю знаний и промежуточной аттестации. Разбор решенных задач, самостоятельное решение задач. | 30 |
| Подготовка к дифференцированному зачету. Повторение теоретического материала по вопросам, совпадающим с темами лекций | 36 |
| **Итого:** | **76** |

# 5. Перечень учебной литературы

***5.1 Основная литература***

1. Касьянов, Виктор Николаевич (1948-). Практикум по программированию : учебное пособие : [для студентов вузов] / В.Н. Касьянов, Е.В. Касьянова ; М-во образования и науки РФ, Новосиб. гос. ун-т, Мех.-мат. фак., Каф. программирования. Новосибирск : Редакционно-издательский центр НГУ, 2013. 197 с. : ил. ; 20 см. URL: <http://e-lib.nsu.ru/dsweb/Get/Resource-949/page001.pdf>. ISBN 978-5-4437-0167-7

***5.2 Дополнительная литература***

1. Лавров, И.А. Задачи по теории множеств, математической логике и теории алгоритмов / И.А. Лавров, Л.Л. Максимова. - 5-е изд., исправл. - Москва : Физматлит, 2002. - 258 с. - ISBN 5-9221-0026-2 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=75576>.
2. Когабаев, Нурлан Талгатович. Лекции по теории алгоритмов [Текст: электронный ресурс] : учебное пособие : [для студентов 1 курса Мех.-мат. фак. НГУ] / Н.Т. Когабаев ; Федер. агентство по образованию, Новосиб. гос. ун-т, Мех.-мат. фак. (Новосибирск : РИЦ НГУ, 2016) . URL: <http://e-lib.nsu.ru/dsweb/Get/Resource-1276/page00000.pdf>.
3. Власов, Владимир Николаевич. Конструирование дискретных математических объектов в парадигме функционального программирования [Текст: электронный ресурс] : учебно-методическое пособие : [для студентов вузов] / В.Н. Власов ; М-во образования и науки РФ, Новосиб. гос. ун-т, Мех.-мат. фак., Каф. дискрет. математики и информатики. (Новосибирск : РИЦ НГУ, 2016) . URL: <http://e-lib.nsu.ru/dsweb/Get/Resource-368/page001.pdf>.

# 6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Для освоения дисциплины используются следующие ресурсы:

- электронная информационно-образовательная среда НГУ (ЭИОС);

- образовательные интернет-порталы;

- информационно-телекоммуникационная сеть Интернет.

Взаимодействие обучающегося с преподавателем (синхронное и (или) асинхронное) осуществляется через личный кабинет студента в ЭИОС.

* 1. ***Современные профессиональные базы данных:***

- электронная библиотека диссертаций Российской государственной библиотеки (ЭБД РГ

- Полнотекстовые журналы Springer Journals за 1997-2015 г., электронные книги (2005-2016 гг.), коллекция научных биомедицинских и биологических протоколов SpringerProtocols, коллекция научных материалов в области физических наук и инжиниринга SpringerMaterials, реферативная БД по чистой и прикладной математике zb

- http://habrahabr.ru - Электронное издание, посвящённое программированию и связанным областям.

***6.2. Информационные справочные системы***

- электронные ресурсы Web of Science Core Collection (Thomson Reuters Scientific LLC.), Journal Citation Reports + ESI

- БД Scopus (Elsevier)

- <http://intuit.ru> - онлайн курсы на русском языке

- <http://coursera.org> - онлайн курсы на английском языке

- <http://projecteuler.net> -сборник задач для самостоятельного освоения языков программирования.

# 7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

***7.1 Перечень программного обеспечения***

Для обеспечения реализации дисциплины «*Декларативное программирование*» используется стандартный комплект программного обеспечения (ПО), включающий регулярно обновляемое лицензионное ПО Windows и MS Office.

# 8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для реализации дисциплины *Декларативное программирование* используются специальные помещения:

1. Учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, лабораторных работ, групповых и индивидуальных консультаций, текущего контроля, промежуточной аттестации;

2. Помещения для самостоятельной работы обучающихся.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду НГУ.

Материально-техническое обеспечение образовательного процесса по данной дисциплине для обучающихся из числа лиц с ограниченными возможностями здоровья осуществляется согласно «Порядку организации и осуществления образовательной деятельности по образовательным программам для инвалидов и лиц с ограниченными возможностями здоровья в Новосибирском государственном университете».

# 9. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине

Перечень результатов обучения по дисциплине *Декларативное программирование* и индикаторов их достижения представлен в виде знаний, умений и владений в разделе 1.

***9.1 Порядок проведения текущего контроля и промежуточной аттестации по дисциплине***

***Текущий контроль успеваемости:***

Текущая аттестация – в форме портфолио.

В состав портфолио входят задания.

В каждом семестре задания делятся на две части: основную и дополнительную.

Для того чтобы получить доступ к дополнительной части заданий, необходимо сдать задания основной части в установленный заранее срок (10я неделя семестра).

Решенные студентом задания должны быть оформлены в виде текста программы и показаны преподавателю.

Предполагается, что студент умеет запустить предоставленный код и ответить на вопросы касательно содержимого.

Таким образом, от студента требуется реализовать необходимую функциональность и ответить на связанные с его реализацией вопросы преподавателя. После этого задание считается сданным.

Отдельные оценки за каждое из заданий не предполагаются.

Основным критерием оценивания является количество сданных заданий.

Если студент сдаёт необходимое количество заданий из базовой части до 10 недели, он получает возможность решать задачи из дополнительной части и претендовать на оценку хорошо и отлично.

Таким образом, сдача основной части даёт оценку «удовлетворительно».

Если студент не сдаёт основную часть заданий вовремя, он не получает возможности претендовать на высокую оценку.

Если студент не сдаёт основную часть в принципе, он не получает зачёта по предмету.

***Промежуточная аттестация:***

Промежуточная аттестация(итоговая по дисциплине) проводится по завершению каждого периода ее освоения (семестра) в форме дифференцированного зачета в 1 семестре и экзамена во 2 семестре.

Промежуточная аттестация проводится в форме дифзачета/экзамена и включает 2 этапа: портфолио и дифзачет/экзамен. Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио. Для оценивания портфолио студенту необходимо сдать все работы, входящие в структуру портфолио.

Дифзачет/экзамен проводится в устной форме. Во время проведения дифзачеа/экзамена студенту разрешается использовать справочники, калькуляторы. В процессе ответа на вопросы билета студенту могут быть заданы дополнительные вопросы по темам дисциплины.

***Описание критериев и шкал оценивания индикаторов достижения результатов обучения по дисциплине*** *Декларативное программирование*

Таблица 9.1

|  |  |  |
| --- | --- | --- |
| **Код компетенции** | **Результат обучения по дисциплине** | **Оценочное средство** |
| ОПК-3 | *Знать принципы и концепции декларативного программирования в части как функционального так и логического подхода, а также основные структуры и методы связанные с декларативными языками.* | Портфолио  Дифзачет/Экзамен |
| *Уметь объяснить характер и порядок вычислений декларативной программы, следуя семантике и прагматике соответствующего языка и виртуальной машины, на которой он реализован.* | Портфолио  Дифзачет/Экзамен |
| *Знать и понимать связи между базисными математическими понятиями (лямбда выражения, типы, классы) и конструкциями языков программирования;*  *Уметь строить иерархические конструкции в языках функционального и логического программирования.* | Портфолио  Дифзачет/Экзамен |

Таблица 9.2

|  |  |
| --- | --- |
| **Критерии оценивания результатов обучения** | **Шкала**  **оценивания** |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все работы, входящие в его структуру (задания делятся на две части: основную и дополнительную).  **Дифференцированный зачет - 1 семестр**  **Экзамен – 2 семестр**  Ответы на вопросы билета.  Студент демонстрирует полное понимание и обоснование теоретических результатов изучаемой темы, приводит примеры из изучаемой темы. Показывает умение применять теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей. Умеет отличать хвостовую рекурсию на практических примерах, умеет по описанию отличить фантомный тип данных.  Умеет интерпретировать функции доступа к компонентам данных (линзы и призмы) Может преобразовать рекурсивную функцию к виду с хвостовой рекурсией, выписать основные алгебраические типы данных, переписать фрагмент кода с использованием линзы Умеет реализовать структу-рированный запрос к дан-ным в форму функциональ-ной программы. Знает концепции декларирования алгоритмов в рекурсивной форме, методы декларации сложных структур данных и представление итеративных программ в функциональной форме с использованием возможностей типовой системы. Знает концепции планирования структурированных запро-сов к данным в форме функциональных программ. | *Отлично* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все работы, входящие в его структуру (задания делятся на две части: основную и дополнительную).  **Дифференцированный зачет - 1 семестр**  **Экзамен – 2 семестр**  Ответы на вопросы билета.  Студент демонстрирует понимание и обоснование теоретических результатов изучаемой темы, частично приводит примеры из изучаемой темы. Показывает умение применять теоретические результаты в различных ситуациях. Умеет распознавать хвостовую рекурсию в практических программах, строить эквивалентную итерационную схему и опознавать фантомный тип данных по описанию. Может преобразовать рекурсивную функцию к виду с хвостовой рекурсией, выписать основные алгебраические типы данных. Знает базисные концепции декларирования алгоритмов в рекурсивной форме, а также методы декларации сложных структур данных. Демонстрирует базовое понимание взаимосвязи функциональной программы и структурированного запроса к данным | *Хорошо* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все основные задания, входящие в его структуру.  **Дифференцированный зачет - 1 семестр**  **Экзамен – 2 семестр**  Ответы на вопросы билета.  Студент демонстрирует частичное понимание изучаемой темы, базовые понятия, алгоритмы, с трудом приводит примеры из изучаемой темы. Частично показывает умение применять теоретические результаты в различных ситуациях. Умеет в учебных задачах распознавать хвостовую рекурсию в практических программах и строить эквивалентную итерационную схему. Может преобразовать рекурсивную функцию к виду с хвостовой рекурсией. Слабо знает основные базисные концепции декларирования алгоритмов в рекурсивной форме. | *Удовлетворительно* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все основные задания, входящие его структуру. Если студент не сдаёт основную часть в принципе, он не получает зачёта по предмету.  **Дифференцированный зачет - 1 семестр**  **Экзамен – 2 семестр**  Студент не может показать знания фактического материала (базовые понятия, алгоритмы, факты) и умение правильно из применять в конкретных заданиях.  Не умеет распознавать хвостовую рекурсию в практических программах и строить эквивалентную итерационную схему. Не знает базисные концепции декларирования алгоритмов в рекурсивной форме.  Не отвечает на дополнительные вопросы. | *Неудовлетво-рительно* |

***Типовые контрольные задания и иные материалы, необходимые для оценки результатов обучения***

|  |  |
| --- | --- |
| Семестр | Формулировка вопроса |
| Семестр 1 | Парадигмы программирования. |
| Основы языка Haskell, типизация данных и функций, работа со списками. |
| Сопоставление с образцом (pattern matching). |
| Хвостовая рекурсия, представление циклов в рекурсивном виде. |
| Виды замыканий и реализация в различных языках программирования. |
| Обобщенные функции и параметрический полиморфизм. |
| Контейнеры и классы типов. |
| Свертки, их варианты (foldl, foldr) и их использование для обработки списков и обобщенных структур. |
| Функции высшего порядка. |
| Стандартные функции обработки списков и списковое включение. |
| Ленивые вычисления и мемоизация. |
| Бесконечные структуры данных. |
| Алгебраические типы данных в языке Haskell. |
| 1. Представление текста в Haskell и его обработка. Регулярные выражения в Haskell |
|  |
| Семестр 2 | Моноиды, функторы и аппликативные функторы. |
| Монады, монадные значения и монадная функция. Класс «Monad». Законы монад. |
| Конкретные монады и их приложения (Maybe, Either, Writer, State и Cont) |
| Графы, деревья, обходы деревьев. Конечные автоматы в функциональном языке, связь с регулярными выражениями. |
| Грамматики, дерево синтаксического разбора, парсеры. Схема восходящего парсера. |
| Шаблоны (Template) в Haskell. |
| Расширения системы типов Haskell. Модель типизации Хиндли-Милнера. |
| Унарные операции реляционной алгебры и SQL |
| Реляционные операции над множествами |
| θ-соединение и его частные случаи |
| Внешнее соединение |
| Группировка и агрегация, порядок вычислений |

Вопросы к дифференцированному зачету и экзамену

Оценочные материалы по промежуточной аттестации (приложение 2), предназначенные для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, хранятся на кафедре-разработчике РПД в печатном и электронном виде.

**Лист актуализации рабочей программы дисциплины**

**«*Декларативное программирование*»**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Характеристика внесенных изменений (с указанием пунктов документа) | Дата и №  протокола Ученого совета ВКИ | Подпись  ответственного |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |