ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ

КАФЕДРА КИБЕРНЕТИКИ

ОДОБРЕНО  
  
протокол № 18 / 03  
  
от « 31 » мая 2018 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**КОНСТРУИРОВАНИЕ ПРОГРАММНЫХ СИСТЕМ**

|  |  |
| --- | --- |
| Направление подготовки (специальность) | 09.04.04 Программная инженерия |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Семестр** | **Интерактив** | **Трудоемкость, кред.** | **Общий объем курса, час.** | **Лекции, час.** | **Практич. занятия, час.** | **Лаборат. работы, час.** | **СРС, час.** | **КСР, час.** | **Форма(ы) контроля, экз./зач./КР/КП** |
| 3 |  | 5 | 180 | 32 | 0 | 32 | 96 | 0 | Э КР |
| ИТОГО | 0 | 5 | 180 | 32 | 0 | 32 | 96 | 0 |  |

Группа: М20-504

АННОТАЦИЯ

Дисциплина «Конструирование программных систем» представляет введение в проектирование сложных программных систем на основе математических моделей вычислений с объектами. Рассматривается аппликативный подход к разработке, в основе которого лежит композиционное построение программ из относительно независимых, замкнутых блоков. При этом выполнение программы рассматривается как протекание объектов данных через структурированные процессы, в результате чего конструируются новые объекты данных, часть из которых играют роль промежуточных результатов и затем отбрасываются, а часть – представляют конечный результат работы (под)программы. При этом рассматриваются как «чистые» вычислительные процессы, так и процессы, в ходе которых вносятся изменения в окружение; однако эти изменения локализуются и явным образом декларируются. В связи с различными стратегиями выполнения вычислительных процессов рассматриваются элементы семантики компьютерных программ.

1. ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Дисциплина «Конструирование программных систем» развивает навыки проектирования программных систем различного типа. При этом дается инструмент для (1) математического анализа корректности тех или иных практических приемов и типовых решений, а также (2) синтеза подходящих решений исходя из особенностей конкретной задачи. Изучение теоретического материала подкрепляется практикой в форме реализации курсового проекта, сбалансированного по сложности задачи и компактности решения.

2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина «Конструирование программных систем» относится к вариативной части профессионального цикла и является дисциплиной по выбору.

Дисциплина в базовом варианте требует предварительной подготовки в объеме курса «Конструирование программных систем» и может рассматриваться как продолжение этого курса в части применения рассмотренных в нем моделей к проектированию и реализации программных систем. При этом изучение теоретических положений подкрепляется практикой в форме реализации нетривиального и многоаспектного, но компактного по формулировке и решению курсового проекта.

Предшествующие дисциплины:

- Методология научных исследований

- Основы автоматизированных информационных технологий

- Функциональное программирование

- Дискретная математика (модели вычислений)

- Формализмы в информационных технологиях

- Семантически безопасное информационное моделирование

В свою очередь дисциплина предоставляет понятийный каркас для изложения методов моделирования информационных и физических процессов, подчеркивая фундаментальную роль информационных процессов в современной картине мира. Рассматриваемые модели вычислений составляют основу для оперирования функциями в технологиях программирования (функциональное и логическое программирование), а также для моделирования динамики предметных областей для систем, основанных на знаниях. Кроме того, дисциплина дает безусловную базу для выполнения научно-исследовательской работы (НИР) в областях прикладной математики и информатики. В особенности, понятийный каркас дисциплины может быть плодотворен для области анализа/разработки/применения информационных систем в Веб.

Параллельные дисциплины:

- Семантическое конфигурирование программных систем

- Абстрактные вычислительные машины

Последующие дисциплины:

- Моделирование (корпоративные информационные системы)

- Теория систем и системный анализ

- Научно-исследовательская работа

Для усвоения курса желательно знакомство с формальными системами и элементами математической логики. Как минимум, необходимо владение представлением об объекте в информатике и о функции в анализе, основами объектно-ориентированного программирования. Более глубокое изучение отдельных элементов курса достигается в дисциплинах: теория типов, семантическое моделирование, концептуальное моделирование и проектирование, модели данных и др.

3. КОМПЕТЕНЦИИ СТУДЕНТА, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ/ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ОБРАЗОВАНИЯ И КОМПЕТЕНЦИИ СТУДЕНТА ПО ЗАВЕРШЕНИИ ОСВОЕНИЯ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОПК-2 – Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач

ПК-12 – Способен проектировать вспомогательные и специализированные языки программирования и языки представления данных

ПК-16 – Владеет навыками создания трансляторов и интерпретаторов языков программирования

ПК-18 – Владеет навыками создания компонент операционных систем и систем реального времени

ПК-19 – Владеет навыками создания систем обработки текстов

- знания:

на уровне представлений: семантика выполнения компьютерных программ; формальный подход к эквивалентным преобразованиям архитектуры приложения и программного кода;

на уровне воспроизведения: семантика простейших императивных возможностей языков программирования; аппликативный, композиционный подход к организации вычислений;

на уровне понимания: построение и организация структур данных; принципы типизации; базовые и производные методы композиции вычислительных процессов; базовые и производные методы конструирования информационных объектов.

- умения:

теоретические: выбор/разработка вычислительной модели, обеспечивающей потребности конкретного приложения; выбор подходящих композиционных схем для вычислительных процессов; синтез подходящих структур данных; выбор режимов выполнения (семантики) вычислительных/информационных процессов;

практические: выработка архитектуры, удовлетворяющей заданной вычислительной модели; анализ ключевых требований, влияющих на архитектуру будущей системы;

навыки: формализация ключевых функциональных, эксплуатационных и других требований; проектирование вычислительных и информационных систем; проектирование вычислительных и информационных сервисов; проектирование и реализация механизмов взаимодействия сервисов.

4. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№ п.п** | **Наименование раздела учебной дисциплины** | **Недели** | **Лекции, час.** | **Практ. занятия / семинары, час.** | **Лабораторные работы, час.** | **Обязат. текущий контроль (форма\*, неделя)** | **Аттестация раздела (форма\*, неделя)** | **Максимальный балл за раздел\*\*** |
|  | *3 Семестр* |  |  |  |  |  |  |  |
| 1 | Система объектов и аппликативные вычисления | 1-4 | 8 | 0 | 8 | КР-4 | КИ-4 | 15 |
| 2 | Конструкции в программных системах | 5-8 | 8 | 0 | 8 | КР-8 | КИ-8 | 15 |
| 3 | Семантика конструкций и вычисление | 9-16 | 16 | 0 | 16 | КР-16 | КИ-16 | 20 |
|  | *Итого за 3 Семестр* |  | 32 | 0 | 32 |  |  | 50 |
|  | **Контрольные мероприятия за 3 Семестр** |  |  |  |  |  | Э КР | 50 |

\* – сокращенное наименование формы контроля

\*\* – сумма максимальных баллов должна быть равна 100 за семестр, включая зачет и (или) экзамен

Сокращение наименований форм текущего контроля и аттестации разделов:

|  |  |
| --- | --- |
| **Обозначение** | **Полное наименование** |
| КИ | Контроль по итогам |
| Э | Экзамен |
| КР | Курсовая работа |

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Недели** | **Темы занятий / Содержание** | **Лек., час.** | **Пр./сем., час.** | **Лаб., час.** |
|  | *3 Семестр* | 32 | 0 | 32 |
| **1-4** | **Система объектов и аппликативные вычисления** | 8 | 0 | 8 |
| 1 - 2 | **Система объектов** Система объектов, строение. Комбинаторная логика. Объекты в комбинаторной логике. -исчисление. Объекты в -исчислении. Понятие комбинатора. Замкнутость, взаимодействие со средой. Комбинаторы в комбинаторной логике и -исчислении. Редукция. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| 3 - 4 | **Типизация объектов** Типизация объектов. Типизация по Черчу. Типизированный язык комбинаторной логики. Типизированный язык -исчисления. Редукция типизированных объектов. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| **5-8** | **Конструкции в программных системах** | 8 | 0 | 8 |
| 5 - 6 | **Понятие об информационном процессе как о вычислении** Понятие об информационном процессе как о вычислении. Средства конструирования процессов, базовые и производные конструкторы. Базовые средства инкапсуляции. Разделяемый код, продвинутые средства инкапсуляции. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| 7 - 8 | **Структуры данных** Структуры данных. Базовые и производные средства конструирования объектов данных. Декартовы произведения. Именованные декартовы произведения. Дизъюнктивные суммы. Списки, коллекции. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| **9-16** | **Семантика конструкций и вычисление** | 16 | 0 | 16 |
| 9 - 10 | **Исполнение вычислительных процессов** Понятие ревизии. Важность раннего выявления ошибок проектирования. Взаимодействие фаз анализа, проектирования и реализации. Виды ревизии. Распределение ролей в группе контроля качества. Цель и методика ревизии. Объекты ревизии. Организация инспекции. Метрики и средства автоматизации ревизии. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| 11 - 12 | **Метод доказательства свойств программы** Метод доказательства свойств программы. Индукция, разбор случаев. Доказательство завершаемости. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| 13 - 14 | **Примеры доказательства свойств программы** Примеры доказательства свойств программы. Типы. | Всего аудиторных часов | | |
| 4 |  | 4 |
| Онлайн | | |
|  |  |  |
| 15 - 16 | **Императивные возможности, классификация** Императивные возможности, классификация. Исключения. Ввод-вывод. Потоки символов. Модифицируемые структуры данных. Семантика деструктивных операций. | Всего аудиторных часов | | |
| 2 |  | 4 |
| Онлайн | | |
|  |  |  |

Сокращенные наименования онлайн опций:

|  |  |
| --- | --- |
| **Обозначение** | **Полное наименование** |
| ЭК | Электронный курс |
| ПМ | Полнотекстовый материал |
| ПЛ | Полнотекстовые лекции |
| ВМ | Видео-материалы |
| АМ | Аудио-материалы |
| Система объектов и аппликативные вычисления | Система объектов и аппликативные вычисления |
| Т | Тесты |
| ЭСМ | Электронные справочные материалы |
| ИС | Интерактивный сайт |

ТЕМЫ ЛАБОРАТОРНЫХ РАБОТ

|  |  |
| --- | --- |
| **Недели** | **Темы занятий / Содержание** |
|  | *3 Семестр* |
| 1 - 4 | **Часть I** **Система объектов и аппликативные вычисления**  Система объектов, строение. Комбинаторная логика. Объекты в комбинаторной логике.  Замкнутость, взаимодействие со средой.  Комбинаторы в комбинаторной логике и -исчислении. Редукция. |
| 5 - 8 | **Часть II Конструкции в программных системах**  Средства конструирования процессов, базовые и производные конструкторы.  Структуры данных.  Базовые и производные средства конструирования объектов данных. |
| 9 - 16 | **Часть III Семантика конструкций и вычисление**  Исполнение вычислительных процессов  Метод доказательства свойств программы. Индукция, разбор случаев. Доказательство завершаемости.  Императивные возможности, классификация.. |

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Методы проведения занятий.

Читаются лекции 2 час в неделю, а также проводятся лабораторные работы 2 часа в неделю. В рамках лабораторных работ изучается решение некоторых типовых задач, а также проводится консультационная работа по выполнению и прием индивидуальных курсовых проектов.

Формы контроля.

Предусмотрена курсовая работа – индивидуальное задание, которое выдается на семестр. Выполнение курсового проекта организуется в три этапа, каждый этап привязан к разделу курса. Прием работы состоит в демонстрации реализации и исходного кода, беседы по теории и вопросов по ходу решения задач. Также в конце каждого раздела предусматривается контрольно-тестовая работа для оценки степени усвоения теоретических знаний и навыков. Это позволяет контролировать как усвоение теоретического материала, так и уровень овладения практическим решением задач. Итоговым контролем является экзамен, включающий ответы на вопросы и решение задач. При определении итоговой оценки учитываются баллы, полученные студентами в семестре: за качество своевременность выполнения курсовых работ.

Технологические особенности

Технологической особенностью изложения дисциплины является отражение лучших мировых практик преподавания подобных курсов в ведущих университетах мира. Студентам предоставляется возможность и необходимая информация для ознакомления с методами и подходами, относящимися к кругу вопросов дисциплины и применяемыми лучшими преподавателями университетов и учебных центров мира.

В частности, для ознакомления, анализа и сопоставления предоставляются ссылки на публично доступный мультимедийный контент и/или электронные формы издания научно-методического материала.

6. ТРЕБОВАНИЯ К ФОНДУ ОЦЕНОЧНЫХ СРЕДСТВ В РАМКАХ РЕАЛИЗУЕМОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

В качестве оценочного средства используется 100 бальная семестровая система, учитывающая посещаемость лекционных занятий, качество и сроки выполнения курсовых проектов. Каждый раздел проходит отдельную аттестацию, включающую оценку работы над соответствующей частью курсового проекта и результаты КТР. Методика оценки, примеры заданий и др. материалы приведены в фонде оценочных средств (ФОС).

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОСНОВНАЯ ЛИТЕРАТУРА:

1. ЭИ O-99 Guide to Discrete Mathematics : An Accessible Introduction to the History, Theory, Logic and Applications, Cham: Springer International Publishing, 2016

2. ЭИ V76 Science Dynamics and Research Production : Indicators, Indexes, Statistical Laws and Mathematical Models, Cham: Springer International Publishing, 2016

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА:

1. И W81 Combinatory logic in programming : Computations with objects through examples and exercises, Wolfengagen V.E., M.: Center JurInfor, 2003

2. 519 В72 Комбинаторная логика в программировании : Вычисления с объектами в примерах и задачах, В. Э. Вольфенгаген, Москва: Центр ЮрИнфоР, 2003

3. 16 В75 Логика : конспект лекций: техника рассуждений, В.Э. Вольфенгаген, М.: Центр ЮрИнфоР, 2004

4. 004 В72 Методы и средства вычислений с объектами : Аппликативные вычислительные системы, В.Э. Вольфенгаген, Москва: JurlnfoR Ltd; ЮрИнфоР-МГУ, 2004

5. 681.3 В72 Категориальная абстрактная машина : Учеб. пособие, Вольфенгаген В.Э., М.: МИФИ, 1993

6. 519 В72 Аппликативные вычисления на основе комбинаторов и лямбда-исчисления : Учеб. пособие, Вольфенгаген В.Э., Гольцева Л.В., М.: МИФИ, 1992

7. 007 И49 Экспертные системы на реляционной основе : Учеб. пособие, Илюхин А.А.,Исмаилова Л.Ю.,Шаргатова Э.И., М.: МИФИ, 1990

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ:

Специальное программное обеспечение не требуется

LMS И ИНТЕРНЕТ-РЕСУРСЫ:

1. ИНТУИТ (http://intuit.ru)

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Специальное материально-техническое обеспечение не требуется

9. ДОПОлНение

Студенты, изучающие дисциплину "Конструирование программного обеспечения. Проектирование и архитектура программных систем", должны обратить внимание на широту и глубину знаний, требуемых для успешного освоения этой дисциплины. При этом на дисциплину отводится сравнительно небольшое количество часов аудиторных занятий. Отсюда следует, что результат успешного освоения дисциплины во многом определяется самостоятельной работой студентов. В связи с этим можно рекомендовать регулярный просмотр сайтов зарубежных и отечественных компаний, занимающихся разработкой программных систем. Подобные ссылки даются на лекциях и практических занятиях. Следует также участвовать в конференциях и семинарах, которые регулярно проводятся в Москве ведущими компаниями-разработчиками программных продуктов. Стоит обратить внимание на следующие сайты:

1) http://events.cnews.ru/events/

2) http://www.ibm.com/ru/events/

3) http://www.academy.it.ru/ru/about/actions/

4) http://www.techdays.ru/

5) https://msevents.microsoft.com/cui/default.aspx?culture=ru-ru

6) http:// h41110.www4.hp.com/ur/institute.html

и многие другие. Многие ведущие программные компании (IBM, MS, HP и др.) реализуют программы академической инициативы, которые позволяют студентам получать бесплатно программные продукты, проходить курсы обучения, участвовать в семинарах и конференциях.

Рабочая программа дисциплины составлена в соответствии с ОС НИЯУ МИФИ (ФГОС) и учебным планом основной образовательной программы (программ).

Автор(ы):

|  |  |  |
| --- | --- | --- |
|  | Исмаилова Лариса Юсифовна, к.т.н. |  |
|  | Рословцев Владимир Владимирович |  |
|  | Вольфенгаген Вячеслав Эрнстович, д.т.н., профессор |  |
|  | Косиков Сергей Владимирович |  |