

Программа зачета
по курсу «Объектно-ориентированное программирование»

1. Введение в ООП. Объектно-ориентированное и структурное программирование: данные и функции, моделирование и кодирование. Эволюция языков и стилей программирования. Объектно-ориентированные языки программирования. Преимущества ООП. Общение в группе разработчиков. Многократное использование.
2. Объектные понятия. Объекты, сообщения, данные и методы, инкапсуляция. Взгляд разработчика и пользователя на объект. Классы. Java-классы: описание, инкапсуляция, экземпляризация, вызов методов. Класс и таблица.
3. Основные принципы ООП. Инкапсуляция, наследование, полиморфизм. Обобщение и конкретизация на UML-диаграммах. Переопределение методов. Реализация наследования в Java.
4. Введение в UML. История UML. UML-диаграммы. Диаграммы вариантов использования, классов, объектов, деятельности, состояний, последовательности, коммуникации. Нотация и средства UML. Использование UML в процессе разработки.
5. Жизненный цикл проекта. Жизненный цикл ПО. Стандарты жизненного цикла (международный и российский). Традиционные фазы жизненного цикла. Характеристики (преимущества, недостатки, условия применения) моделей жизненного цикла: каскадной, спиральной, инкрементной (приращений), итерационной. Жизненный цикл объектно-ориентированной разработки и наиболее подходящие для него модели.
6. Варианты использования и требования системы. Применение и описание варианта использования. Идентификация акторов и вариантов использования. Пред- и постусловия. Полное описание варианта использования. Сбор технических требований. Варианты использования в жизненном цикле проекта.
7. Обзор Java-технологии. История и задачи Java-технологии. Элементы Java-платформы. Виртуальная машина Java (JVM). Библиотеки стандартных классов. Выпуски Java-платформы. Хранение Java-классов. Компиляция и запуск. Функции JVM: динамическая загрузка, создание объектов, управление памятью, сборка мусора. JavaBeans. Взаимодействие Java-программ с окружением. Java EE: многоуровневые приложения, контейнеры, стандарты и API, Web-технологии. Компоненты Java-приложений. Архивные файлы Java. Эволюция Java-платформы.
8. Введение в Java-язык. Структура Java-программы. Пакеты. Базовая грамматика: синтаксис и именованность. Предложения, блоки и классы. Абстрактные типы данных. Класс java.lang.Object. Объявление полей, методов и конструкторов. Описание конструкторов. Перегрузка конструкторов и методов. Объявление переменных экземпляра и класса, методов классов, локальных переменных. Видимость и доступность переменных. Простые (примитивные) типы данных, приведение типов, операторы. Объектные указатели. Класс String, синтаксис, операторы. Сравнение переменных. Составные (агрегированные) типы: строки, массивы, перечисления, коллекции. Классы-обертки. Консоль ввода/вывода. Условные операторы. Циклы. Точки входа в Java-программу. Метод main().
9. Объектно-ориентированный анализ. Статические UML-диаграммы. Переход от сбора требований к анализу. Статическая и динамическая модель системы. Методики идентификации объектов. Диаграмма классов. Актор и приложение как

объекты. Управление выполнением. Классы проблемной области и классы-утилиты.

10. Объектно-ориентированный анализ. Динамические UML-диаграммы. Реализация варианта использования. Представление сценария на диаграммах последовательности, состояний, деятельности, коммуникации.
11. Объектно-ориентированное проектирование для реализации. Ассоциации. Идентификация классов и свойств. Статические и динамические отношения. Идентификация и классификация ассоциаций. Представление ассоциаций на диаграмме классов. Характеристики, направленность и множественность ассоциаций. Проектирование и реализация ассоциаций в качестве свойств. Варианты хранения ассоциаций. Оптимизация проектирования ассоциаций. Модель системы на этапе проектирования.
12. Объектно-ориентированное проектирование для реализации. Наследование. Наследование на этапе проектирования. Отображение наследования на диаграмме классов. Реализация наследование в Java. Многократное использование. Иерархия наследования на примере коллекций. Переопределение методов. Абстрактные методы и классы. Конечные классы. Динамическое связывание в качестве полиморфизма. Интерфейсы. Пример интерфейсов в коллекциях. Отображение интерфейсов на диаграмме классов. Доступ и приведение типов в иерархиях наследования.
13. Проектирование для изменений. Изменения и необходимость их поддерживать. Изменения на уровне классов и их влияние на остальные классы. Связанность между классами и пакетами и связанность внутри них. Архитектурная модуляризация. Пример многоуровневой архитектуры. Паттерны проектирования. Порождающие паттерны Singleton и Factory Method. Поведенческий паттерн Command. Архитектурный паттерн MVC, компоненты, реализация в Java.
14. Методологии. Rational Unified Process, фазы, визуальное моделирование. Гибкая (Agile) методология, манифест, методы. Сравнение RUP и Agile.

Список основной литературы

1. Объектно-ориентированный анализ и проектирование с примерами приложений / Г. Буч [и др.]. 3-е изд. М.: Вильямс, 2008. 718 с.
2. Буч Г., Якобсон А., Рамбо Дж. UML. 2-е изд. СПб.: Питер, 2006. 735 с. (шифр для библиотеки МИРЭА: 004 Б94)
3. Эккель Б. Философия Java. 4-е изд. СПб.: Питер, 2009. 637 с.
4. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма [и др.]. 1-е изд. Питер, 2007. 366 с. (шифр для библиотеки МИРЭА: 004 П75)

Список дополнительной литературы

5. Рамбо Дж., Блаха М. UML 2.0. Объектно-ориентированное моделирование и разработка. СПб.: Питер, 2007. 544 с. (шифр для библиотеки МИРЭА: 004 P21)
6. Арлоу Дж., Нейштадт А. UML 2 и Унифицированный процесс. 2-е изд. М.: Символ, 2008. 621 с.
7. Ноутон П., Шилдт Г. Java 2. Наиболее полное руководство. БХВ-Петербург, 2008. 1072 с. (шифр для библиотеки МИРЭА: 004 H85)
8. <http://www.uml.org/>
9. <http://www.eclipse.org/>
10. <http://java.sun.com/>