**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования
"Национальный исследовательский университет
"Высшая школа экономики"**

Факультет Бизнес информатики

**Программа дисциплины**

Компонентно-ориентированное программирование

для направления 231000.62 Программная инженерия

подготовки бакалавра

Автор программы:

Ланин В.В., lanin@perm.ru

Одобрена на заседании кафедры информационных технологий в бизнесе

«\_\_»\_\_\_\_\_\_\_\_\_\_ 2013 г

И.о. зав. кафедрой О.Л. Викентьева \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Утверждена Учебно-методическим Советом НИУ ВШЭ – Пермь

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2013 г.

Пермь, 2013

*Настоящая программа не может быть использована другими подразделениями университета и другими вузами без разрешения кафедры-разработчика программы.*

# Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает минимальные требования к знаниям и умениям студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих данную дисциплину, учебных ассистентов и студентов направления подготовки 231000.62 Программная инженерия, изучающих дисциплину «Компонентно-ориентированное программирование».

Программа разработана в соответствии с:

* Образовательным стандартом федерального государственного автономного образовательного учреждения высшего профессионального образования Национального исследовательского университета «Высшая школа экономики» по направлению подготовки 231000.62 Программная инженерия (уровень подготовки бакалавр). Утверждён 02.07.2010 (протокол № 15), редакция 2011 г.
* Учебным планом университета по направлению подготовки 231000.62 Программная инженерия, утвержденным в 2012 г.

# Цели освоения дисциплины

Целями освоения дисциплины «Компонентно-ориентированное программирование» являются:

*В области обучения* – получение высшего профессионально профилированного (на уровне бакалавра) образования, позволяющего выпускнику успешно работать в избранной сфере деятельности, обладать универсальными и предметно-специализированными компетенциями, способствующими его социальной мобильности и устойчивости на рынке труда, а именно: изучение технологий, используемых в конструировании программных систем, создаваемых на принципах открытых систем с использование технологий COM, .NET и JAVA.

*В области воспитания* – развитие у студентов социально-личностных качеств: целеустремленности, организованности, трудолюбия, ответственности, самостоятельности, гражданственности, эмоционального интеллекта, приверженности этическим ценностям, коммуникативности, толерантности, повышение их общей культуры и мышления.

# Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате изучения дисциплины студент должен:

* **знать**
	+ основные принципы разработки программных компонентов;
	+ компонентные технологий COM, CORBA, .NET;
* **уметь**
	+ применять компонентный подход при разработке программ;
* **приобрести навыки**
	+ компонентного программирования на языках C++, C#, JAVA;
	+ развертывания и администрирования программных систем, построенных на основе компонентного подхода.

В результате освоения дисциплины студент осваивает следующие компетенции:

| **Компетенция** | **Код по ФГОС/ НИУ** | **Дескрипторы – основные признаки освоения (показатели достижения результата)** | **Формы и методы обучения, способствующие формированию и развитию компетенции** |
| --- | --- | --- | --- |
| Способность формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта. | ПК-6 | Студент способен после этапа анализа области и требований к системе разработать архитектуру системы, выделив основные компоненты и выполняемые ими функции. | Аудиторные занятия проводятся в форме, предусматривающей участие студентов в обсуждении и анализе проблем и методов решений. |
| Способность создавать программные интерфейсы. | ПК-14 | Студент способен проектировать программные интерфейсы, обеспечивающие расширяемость и масштабируемость системы. | Самостоятельное изучение отдельных тем. Выполнения индивидуальных заданий (с получением консультаций преподавателя) |
| Навыки использования различных технологий разработки программного обеспечения | ПК-16 | Студент имеет опыт использования технологий .NET и JAVA | Выполнение заданий, связанных с разработкой офисных приложений |
| Понимание особенностей эволюционной деятельности, как с технической точки зрения, так и с точки зрения бизнеса (работа с унаследованными системами, возвратное проектирование, реинженеринг, миграцию и рефакторинг). | ПК-27 | Студент знает основные методики и способен на практике решать задачи работы с унаследованными системами, возвратного проектирования, реинженеринга, миграции и рефакторинга. | Аудиторные занятия проводятся в форме, предусматривающей участие студентов в обсуждении и анализе проблем и методов решений. |
| Умение применять основные методы и инструменты разработки программного обеспечения. | ПК-17 | Студент способен подбирать и применять инструментальные и программные средства для решения поставленных задач. | Аудиторные занятия проводятся в форме, предусматривающей участие студентов в обсуждении и анализе проблем и методов решений. |

# Место дисциплины в структуре образовательной программы

Настоящая дисциплина относится к циклу дисциплин «Факультативы».

Изучение данной дисциплины базируется на следующих дисциплинах:

* Программирование
* Информатика, математическая логика и теория алгоритмов
* Введение в программную инженерию.
* Конструирование программного обеспечения.

Для освоения учебной дисциплины, студенты должны владеть следующими знаниями и компетенциями:

* Знание основ функционирования персональных компьютеров.
* Знание основ процедурного и объектно-ориентированного программирования.
* Навыки работы в среде Microsoft Windows.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин:

* Конструирование программного обеспечения (продолжение курса).
* Введение в методы трансляции.
* Проектирование и архитектура программных систем.
* Распределенные вычисления.
* Человеко-машинный интерфейс.

# Тематический план учебной дисциплины

|  |  |  |  |
| --- | --- | --- | --- |
| **Названия разделов и тем** | **Всего часов** | **Аудиторные часы** | **самостоя­тельная работа** |
| **Лекции** | **Семинары** | **Практические занятия** |
| **Раздел 1.****Введение в компонентные технологии** |
| 1. Проблемы разработки сложных программных систем
 | 2 | 2 |  |  |  |
| 1. Теоретические основы компонентного подхода
 | 2 | 2 |  |  |  |
| 1. Понятие распределенной системы
 | 2 | 2 |  |  |  |
| 1. Паттерны проектирования
 | 7 | 2 |  | 4 | 1 |
| 1. Web сервисы
 | 7 | 2 |  | 4 | 1 |
| **Раздел 2. Компонентные технологии Microsoft** |
| 1. Модель COM
 | 7 | 2 |  | 4 | 1 |
| 1. Развитие технологии COM
 | 7 | 2 |  | 2 | 1 |
| 1. Платформа Microsoft .NET
 | 10 | 4 |  | 4 | 2 |
| 1. Очереди сообщений MSMQ
 | 4 | 2 |  | 2 | 0 |
| 1. Microsoft .NET WCF
 | 10 | 4 |  | 4 | 2 |
| **Раздел 3. Другие технологии построения распределенных систем** |
| 1. Стандарты CORBA
 | 6 | 2 |  | 2 | 2 |
| 1. Технология JavaBeans
 | 8 | 4 |  | 4 | 2 |
| **ИТОГО:** | **72** | **30** |  | **30** | **12** |

# Контроль знаний студентов

|  |  |  |  |
| --- | --- | --- | --- |
| **Тип контроля** | **Форма контроля** | **1 год** | **Параметры** |
| **1** | **2** | **3** | **4** |
| Текущий (неделя) | Домашнее задание № 1 | 5 |  |  |  | Разработка программного компонента с использованием технологии COM. |
| Домашнее задание № 2 |  | 5 |  |  | Разработка программного компонента с использованием технологии JavaBeans. |
| Итоговый | Зачет |  | \* |  |  | Итоговый тест с закрытыми и открытыми ответами |

## Критерии оценки знаний, навыков

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

Критерии оценки выполнения заданий:

|  |  |
| --- | --- |
| **Характеристика решения** | **Оценка** |
| Задание выполнено полностью, студент объясняет решение, реализованы дополнительные функции приложения  | 10 |
| Задание выполнено полностью, студент объясняет решение | 9 |
| Задание выполнено полностью, студент объясняет решение, но затрудняется в ответе на некоторые вопросы | 8 |
| Задание выполнено полностью, но в процессе работы программы возникают ошибки времени выполнения | 6‑7 |
| Задание выполнено на 70 %, в процессе работы программы возникают ошибки времени выполнения | 4‑5 |
| Задание выполнено на 50 %, в процессе работы программы возникают ошибки времени выполнения | 3 |
| Задание выполнено на 25 %, в процессе работы программы возникают ошибки времени выполнения | 2 |
| Задание выполнено на 10 %, в процессе работы программы возникают ошибки времени выполнения | 1 |
| Задание не выполнено | 0 |

## Порядок формирования оценок по дисциплине

Преподаватель оценивает работу студентов на практических занятиях и выполнение домашних заданий.

При определении оценки учитываются:

* Качество выполнения работ на практических занятиях,
* Ответы на контрольные вопросы по работам на практических занятиях,
* Скорость выполнения работ,
* Креативность в решении поставленных проблем.

Оценки за работу на семинарских и практических занятиях преподаватель выставляет в рабочую ведомость. Оценка по 10-ти балльной шкале за работу на семинарских и практических занятиях определяется перед итоговым контролем и называется - *Оаудиторная*.

**Оценка за текущий контроль** *(Отекущий*) рассчитывается как взвешенная сумма всех форм текущего контроля.

*Отекущий* = *n1·Од/з1 + n2·Од/з2*,

при этом n1 = 0,5, n2 = 0,5.

Способ округления оценки за текущий контроль: арифметический.

**Накопленная оценка** за текущий контроль учитывает результаты студента по текущему контролю следующим образом:

О*накопленная*= 2/3\* *Отекущий* + 1/3\* О*аудиторная .*

Способ округления накопленной оценки текущего контроля: арифметический.

**Результирующая оценка** за дисциплину рассчитывается следующим образом

*Орезультирующая = 0,6\* Онакопленная + 0,4\*·Озач*

Способ округления накопленной оценки промежуточного (итогового) контроля в форме зачета: арифметический.

На пересдаче студенту не предоставляется возможность получить дополнительный балл для компенсации оценки за текущий контроль.

На зачете студент может получить дополнительную практическую задачу, ответ на который оценивается в 1 балл.

В диплом выставляет результирующая оценка по учебной дисциплине, которая формируется равной результирующей оценке (*Орезультирующая*).

# Содержание дисциплины

1. Раздел 1. Введение в компонентные технологии.

**Тема 1.** Проблемы разработки сложных программных систем

Понятие программной инженерии. Программные системы. Программные комплексы. Распределенные программные системы. Проблемы разработки приложений. Прагматические принципы работы со сложными системами: абстракция и уточнение, модульная разработка и повторное использование программного кода.

Количество часов аудиторной работы: 2 часа.

Общее количество часов самостоятельной работы: 0 часов.

**Тема 2.** Теоретические основы компонентного подхода

Архитектура программного обеспечения. Разработка и оценка архитектуры на основе сценариев. Парадигма компонентного программирования. Философия и методология компонентного подхода. Программный модуль. Компонент. Архитектурный компонент. Компонентная модель. Интерфейсы. Адекватность, полнота, минимальность и простота интерфейсов.

Понятие стандарта компонентного программирования: использование компонентов, требования к компонентам.

Синхронное и асинхронное взаимодействие. Очередь сообщений. Маршалинг. Сериализация. Реализация асинхронного взаимодействия при помощи очередей сообщений.

Транзакции. Свойства транзакции. Протокол двухфазного подтверждения транзакции.

Количество часов аудиторной работы: 2 часа.

Общее количество часов самостоятельной работы: 0 часов.

**Тема 3.** Понятие распределенной системы

Общие принципы построения распределенных систем: прозрачность, открытость, масштабируемость, безопасность. Задачи, решаемые в рамках создания распределенных систем: организация связи и передачи данных между элементами системы, поддержка идентификации и поиска отдельных ресурсов внутри системы, организация работ в рамках процессов и потоков, синхронизация параллельно выполняемых потоков работ, поддержка целостности данных и непротиворечивости вносимых изменений, обеспечение отказоустойчивой работы, защита данных и коммуникаций.

Количество часов аудиторной работы: 2 часа.

Общее количество часов самостоятельной работы: 0 часов.

**Тема 4.** Паттерны проектирования

Понятие паттерна проектирование. Описание паттернов проектирования. Порождающие паттерны. Структурные паттерны. Паттерны поведения.

Количество часов аудиторной работы: 6 часов.

Общее количество часов самостоятельной работы: 1 час.

**Тема 5.** Web сервисы

Web сервисы: модель программирования Web сервисов (ASP.NET, Web Forms), описание Web сервисов – WSDL (Web Service Description Language - язык описания Web сервисов), поиск Web сервисов – спецификация UDDI (Universal Description, Discovery and Integration - универсальное описание, поиск и интеграция), протоколы и методы передачи данных - SOAP (Simple Object Access Protocol - простой протокол доступа к объектам), HTTP-Get, HTTP-Post.

Количество часов аудиторной работы: 6 часов.

Общее количество часов самостоятельной работы: 1 час.

**Литература по разделу:**

1. Кулямин В.В. Технологии программирования. Компонентный подход. – М.: Интернет-Университет Информационных Технологий, 2007. – 464 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.
2. Троэлсон Э. Язык программирования С# 5.0 и платформа .NET 4.5 = Pro C# 5.0 and the.NET 4.5 Framework: пер. с англ. – М.: Вильямс, 2013. – 1311 с.
3. Павловская Т.А. C/C++. Программирование на языке высокого уровня: учебник для магистров и бакалавров. – СПб.: ПИТЕР, 2013. – 460 с.
4. Добрынин В.Ю. Технологии компонентного программирования: Учеб. пособие. – СПб.: Издательство С.-Петербургского университета, 2004.
5. Раздел 2. Компонентные технологии Microsoft

**Тема 6.** Модель COM

Компонентная модель объектов COM и ее промышленная реализация. Основные понятия и элементы: язык IDL и базовые интерфейсы. GUID, библиотека типов и системный реестр, моникеры, создание, использование и уничтожение компонента, включение и агрегирование, фабрики классов, модели многопоточности.

Хранилища компонентов. Вопросы связанные с экспонированием фабрик классов, управлением временем жизни, регистрацией сервера.

Разработка элементов управления ActiveX. Элементы управления и страницы свойств. Структура стандартного элемента управления ActiveX. Использование мастера ATL. Элементы управления и категории COM. Постоянства свойств. Код стороны клиента. Пример разработки элемента управления ActiveX с помощью ATL.

Количество часов аудиторной работы: 6 часов.

Общее количество часов самостоятельной работы: 1 час.

**Тема 7.** Развитие технологии COM

Технологии COM+ и DCOM. Службы COM+. Модели обращения к объектам. События. Интеграция с Active Directory. Реализация защиты. Администрирование COM+. Отладка и развертывание приложений COM+.

Количество часов аудиторной работы: 4 часа.

Общее количество часов самостоятельной работы: 1 час.

**Тема 8.** Платформа Microsoft .NET

Common Language Runtime: .NET Framework Class Library, сборки (assemblies).

Количество часов аудиторной работы: 8 часов.

Общее количество часов самостоятельной работы: 2 часа.

**Тема 9.** Очереди сообщений MSMQ

Понятие очереди сообщений. Преимущества использования очередей сообщений. Типы очередей.

Программирование MSMQ: создание очередей, доступ к очередям, удаление очередей, обращение к очередям, посылка сообщения очередям, чтение сообщений очереди, удаление сообщений из очереди.

Количество часов аудиторной работы: 4 часа.

Общее количество часов самостоятельной работы: 0 часов.

**Тема 10.** Microsoft .NET WCF

Контракты сервисов и их реализация. Размещение WCF сервисов. Поведение WCF сервисов. Варианты размещения WCF сервисов. Обзор класса ServiceHost. Размещение WCF сервисов в службах Windows. Размещение в IIS, WAS и AppFabric. Конфигурирование хост приложения.

Введение в понятие контракт. Обзор типов контрактов. Механизмы сообщений. Разработка WCF контракта.

Реализация множественных точек доступа и совместимость. Поиск WCF сервисов. Точки доступа для WCF по умолчанию. Механизмы создания объектов WCF и сценарии использования в многопоточной среде.

Модель безопасности WCF сервисов. Безопасность сообщений и транспортная безопасность. Аутентификация и авторизация. Аутентификация с использованием утверждений (claim-based).

Количество часов аудиторной работы: 8 часов.

Общее количество часов самостоятельной работы: 2 часа.

**Литература по разделу:**

1. Кулямин В.В. Технологии программирования. Компонентный подход. – М.: Интернет-Университет Информационных Технологий, 2007. – 464 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.
2. Троэлсон Э. Язык программирования С# 5.0 и платформа .NET 4.5 = Pro C# 5.0 and the.NET 4.5 Framework: пер. с англ. – М.: Вильямс, 2013. – 1311 с.
3. Павловская Т.А. C/C++. Программирование на языке высокого уровня: учебник для магистров и бакалавров. – СПб.: ПИТЕР, 2013. – 460 с.
4. Макаров Ф.В., Скоробогатов С.Ю., Чеповский A.М. Common Intermediate Language и системное программирование в Microsoft .NET: учебное пособие. – М.: Интернет-Университет Информационных Технологий, 2006. – 328 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.
5. Мюллер Дж. Технология COM+: библиотека программиста. – СПб.: Питер, 2002.
6. Причард Д. COM CORBA. Архитектура, стратегии и реализации. – М.: ЛОРИ, 2001.
7. Раздел 3. Компонентные технологии JAVA

**Тема 11.** Технология CORBA

Спецификация OMA (Object Management Architecture - архитектура управления объектами).

IDL. CORBA-объекты и серванты. Объектные адаптеры. GIOP и IIOP. Механизм выполнения удаленных вызовов.

Сервисы CORBA: жизненного цикла, долговременного хранения, именования, событий, контроля совместного доступа, транзакций, отношений, внешнего представления, запросов, лицензирования, свойств, времени, безопасности, коммерции, контейнеров.

Количество часов аудиторной работы: 4 часа.

Общее количество часов самостоятельной работы: 2 часа.

**Тема 12.** Технология JavaBeans

Основные понятия: сервер IJB, контейнер IJB, компонент IJB, классы и интерфейсы компонента IJB, дескриптор развертывания. Вспомогательные интерфейсные объекты. Установка элемента в контейнер. Виды и атрибуты транзакций.

Количество часов аудиторной работы: 8 часов.

Общее количество часов самостоятельной работы: 2 часа.

**Литература по разделу:**

1. Кулямин В.В. Технологии программирования. Компонентный подход. – М.: Интернет-Университет Информационных Технологий, 2007. – 464 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.
2. Причард Д. COM CORBA. Архитектура, стратегии и реализации. – М.: ЛОРИ, 2001.

# Образовательные технологии

Лекции проводятся с использованием компьютерного мультимедийного оборудования. Предусматривается разбор практических задач.

На практике используется программное обеспечение, позволяющее получить практические навыки работы с инструментальными средствами.

## Методические рекомендации преподавателю

На лабораторных занятиях используются следующие методы обучения и контроля усвоения материала:

1) выполнение лабораторных работ по теме занятия сопровождается контрольным опросом;

2) обсуждение различных вариантов решения, предложенных студентами, сравнение решений, анализ возможных ситуаций.

Методические рекомендации по выполнению лабораторных работ приведены в рекомендованных книгах (электронных ресурсах).

## Методические указания студентам

Студенту рекомендуется следующая схема подготовки к лабораторному занятию:

1) проработать конспект лекций;

2) проанализировать основную и дополнительную литературу, рекомендованную по изучаемому разделу;

3) проанализировать варианты решений, предложенные преподавателем на практических занятиях;

4) при затруднениях сформулировать вопросы к преподавателю.

Подробные методические рекомендации по выполнению лабораторных работ приведены в книгах, указанных в основном списке литературы по курсу (доступны в электронном виде).

# Оценочные средства для текущего контроля и аттестации студента

## Тематика заданий текущего контроля

Тема домашнего задания №1: Разработка программного компонента с использованием технологии COM.

Тема домашнего задания №2: Разработка программного компонента с использованием технологии JavaBeans.

## Вопросы для оценки качества освоения дисциплины

1. Перечислите основные недостатки процедурного программирования.
2. Назовите и охарактеризуйте основные принципы ООП.
3. Парадигма компонентного программирования.
4. Понятие стандарта компонентного программирования.
5. Перечислите требования к компонентам.
6. Понятие паттерна проектирование.
7. Назовите известные вас порождающие паттерны.
8. Назовите известные вас структурные паттерны.
9. Назовите известные вас паттерны поведения.
10. Понятие интерфейса. Преимущества использования интерфейсов.
11. Дайте определение понятия сериализация.
12. Дайте определение транзакции.
13. Перечислите свойства транзакции.
14. Опишите протокол двухфазного подтверждения транзакции.
15. Опишите принципы построения распределенных систем: прозрачность, открытость, масштабируемость, безопасность.
16. Компонентная модель объектов COM. Основные понятия COM. Интерфейсы в COM.
17. Перечислите и охарактеризуйте базовые интерфейсы COM.
18. Назовите основные функции Common Language Runtime.
19. Назовите основные характеристики сборки (assemblies).
20. Перечислите основные этапы динамической загрузки сборок.
21. Опишите основное отличие синхронного и асинхронного взаимодействия.
22. Опишите процесс реализации асинхронного взаимодействия при помощи очередей сообщений.
23. Перечислите основные операции с очередями MSMQ.
24. Опишите два варианта взаимодействия с объектами, расположенными на сервере.

# Учебно-методическое и информационное обеспечение дисциплины

## Базовый учебник

1. Кулямин В.В. Технологии программирования. Компонентный подход. – М.: Интернет-Университет Информационных Технологий, 2007. – 464 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.

## Основная литература

1. Троэлсон Э. Язык программирования С# 5.0 и платформа .NET 4.5 = Pro C# 5.0 and the.NET 4.5 Framework: пер. с англ. – М.: Вильямс, 2013. – 1311 с.
2. Павловская Т.А. C/C++. Программирование на языке высокого уровня: учебник для магистров и бакалавров. – СПб.: ПИТЕР, 2013. – 460 с.

## Дополнительная литература

1. Макаров Ф.В., Скоробогатов С.Ю., Чеповский A.М. Common Intermediate Language и системное программирование в Microsoft .NET: учебное пособие. – М.: Интернет-Университет Информационных Технологий, 2006. – 328 с. [Электронный ресурс]. – URL: http://www.biblioclub.ru.
2. Секунов Н. Разработка приложений на С++ и C#. – СПб.: Питер, 2003.
3. Мюллер Дж. Технология COM+: библиотека программиста. – СПб.: Питер, 2002.
4. Шилдт Г. Полный справочник по C#. – М.: Издательский дом «Вильямс», 2007.
5. Маклин С., Нафтел Д., Уильямс К. Microsoft .NET Remoting – М.:Издательско-торговый дом "Русская редакция", 2003.
6. Причард Д., COM CORBA. Архитектура, стратегии и реализации. – М.: ЛОРИ, 2001.
7. Хильер С. Создание приложений COM+ в среде Visual Basic. Руководство разработчика. – М.: Издательский дом «Вильямс», 2001.
8. Шилдт Г. C#: учебный курс. - СПб.: Питер, 2003.
9. Добрынин В.Ю. Технологии компонентного программирования: Учеб. пособие. – СПб.: Издательство С.-Петербургского университета, 2004.

## Справочники, словари, энциклопедии

Не предусмотрены.

## Программные средства

Для успешного освоения дисциплины, студент использует следующие программные средства:

* Microsoft Visual Studio 2010;
* Eclipse Standard 4.3.2;
* Acrobat Reader.

## Дистанционная поддержка дисциплины

LMS – основа для дистанционной поддержки дисциплины, используется для консультирования по выполнению домашних работ и подготовке к зачёту.

# Материально-техническое обеспечение дисциплины

Для проведения лекционных занятий используется компьютер с установленным программным обеспечением для демонстрации презентаций и проектор.

Практические занятия проводятся в компьютерных классах с установленным программным обеспечением, перечисленным выше.