

**Автономная некоммерческая организация дополнительного
профессионального образования «Академия бизнеса и инновационных
технологий»**



«УТВЕРЖДАЮ»

Директор

Никишина О.Ю.

«02» октября 2017 г.

**Рабочая программа
дисциплины**

«Технология разработки программного обеспечения»

Дополнительная профессиональная программа
по переподготовке

«Системное администрирование и информационные технологии»

Квалификация выпускника
Специалист по информационным системам

Форма обучения
очно-заочная, заочная

Москва, 2017

Составитель (и): Никишин Сергей Анатольевич

Рецензент (ы): Ольховиков Леонид Александрович

Рассмотрена и одобрена на заседании Педагогического совета
Протокол № 1/ПС от 02.10.2017

1. Цели и задачи освоения дисциплины

Цель реализации программы состоит в формировании профессиональных компетенций обучающихся в области освоения различных методов абстрагирования, обеспечения модульности и других аспектов проектирования программных систем.

Задачи:

- ознакомление с современными языками программирования, их классификацией и областями их применения;
- привить навыки абстрагирования, обеспечения модульности и других аспектов проектирования программных систем;
- повышение профессиональной эрудиции.

2. Место дисциплины в структуре ДПП

Данная дисциплина входит в совокупность дисциплин необходимых для подготовки слушателя к выполнению нового вида профессиональной деятельности в области администрирования информационных систем.

Для освоения дисциплины необходимы компетенции, сформированные у обучающихся в результате изучения таких дисциплин как «Информатика», «Математическая логика и теория алгоритмов», «Операционные системы», «Базы данных», «Проектирование автоматизированных информационных систем», «Методы и средства защиты компьютерной информации».

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины формируются следующие профессиональные компетенции:

- готовность к разработке моделирующих алгоритмов и реализации их на базе языков и пакетов прикладных программ моделирования (ПК-3);
- готовность к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ (ПК-5).

В результате освоения дисциплины слушатель должен:

•Знать:

- методы проектирования программных средств вычислительной техники;
- жизненный цикл программ, оценку качества программных продуктов, технологии разработки программных комплексов, CASE-средства;
- методы и алгоритмы объектно-ориентированного программирования.

•Уметь:

- разрабатывать архитектуру программного обеспечения;
- проектировать интерфейс пользователя;
- использовать среды разработки объектно-ориентированных программ;

– выполнять отладку программных продуктов.

• Владеть:

- методиками проектирования и разработки программных систем.

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 42 часа.

4.1. Объем дисциплины и виды учебной деятельности

Очно-заочная форма обучения

Вид учебной работы	Всего часов
Общая трудоемкость дисциплины	42
Аудиторные занятия:	32
Лекции (Л)	10
Практические занятия (ПЗ)	22
Самостоятельная работа	10
Вид итогового контроля	Зачет

Заочная форма обучения с применением дистанционных технологий

Вид учебной работы	Всего часов
Общая трудоемкость дисциплины	42
Аудиторные занятия:	22
Лекции (Л)	10
Практические занятия (ПЗ)	12
Самостоятельная работа	20
Вид итогового контроля	Зачет

4.2. Разделы дисциплины и виды занятий

Очно-заочная форма обучения

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу и трудоемкость (в часах)			Формы текущего контроля успеваемости
		лекции	Практические занятия/ семинары	самостоятельная работа	
1	Жизненный цикл программного обеспечения	2	4	2	-
2	Объектно-ориентированный анализ	2	4	2	-

3	Объектно-ориентированное проектирование программной системы	2	4	2	-
4	Кодирование объектно-ориентированных систем	2	4	2	-
5	Тестирование и отладка программных систем	2	6	2	-
ВСЕГО часов		10	22	10	Зачет

Заочная форма обучения с применением дистанционных технологий

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу и трудоемкость (в часах)			Формы текущего контроля успеваемости
		лекции	Практические занятия/семинары	самостоятельная работа	
1	Жизненный цикл программного обеспечения	2	-	4	-
2	Объектно-ориентированный анализ	2	4	4	-
3	Объектно-ориентированное проектирование программной системы	2	4	4	-
4	Кодирование объектно-ориентированных систем	2	4	4	-
5	Тестирование и отладка программных систем	2	-	4	-
ВСЕГО часов		10	12	20	Зачет

4.3. Тематическое содержание дисциплины

Тема 1. Жизненный цикл программного обеспечения.

Жизненный цикл программного обеспечения. Распределение финансовых и временных затрат на реализацию каждого из этапов разработки программного обеспечения.

Тема 2. Объектно-ориентированный анализ.

Абстрагирование. Теории классификации. Методы выявления классов и объектов.

Тема 3. Объектно-ориентированное проектирование программной системы.

Построение объектно-ориентированной архитектуры системы. Методы объектно-ориентированного анализа для выявления классов и объектов. CASE-средства объектно-ориентированного проектирования.

Тема 4. Кодирование объектно-ориентированных систем.

Современные технологии объектно-ориентированного программирования. Компонентный подход к созданию объектно-ориентированных систем. Технология.NET.

Тема 5. Тестирование и отладка программных систем.

Стратегии и методы тестирования. Прямое и обратное тестирование. Программные средства автоматизации тестирования.

5. Образовательные технологии, применяемые при освоении дисциплины

Для обеспечения качественного образовательного процесса применяются следующие образовательные технологии:

- традиционные: лекции, практические занятия;
- инновационные: интерактивные лекции (в режиме on-line и /или off-line);
- интерактивные: вебинары, чат, форумы, интернет-конференции;
- самостоятельная работа слушателей.

К интерактивным методам относятся также презентации с использованием различных вспомогательных средств: интерактивной доски, раздаточных материалов, видеофильмов, слайдов, мультимедийной презентации и т.п.

Интерактивные методы поощряют активное участие каждого в учебном процессе;

- способствуют эффективному усвоению учебного материала;
- оказывают многоплановое воздействие на обучающихся;
- осуществляют обратную связь (ответная реакция аудитории);
- формируют у обучающихся мнения и отношения; формируют жизненные навыки;
- способствуют изменению поведения.

6. Учебно-методическое обеспечение самостоятельной работы слушателей. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Основными видами самостоятельной работы являются:

- выполнение индивидуальных заданий,
- изучение основной и дополнительной литературы,
- поиск и сбор информации по дисциплине в периодических печатных и интернет-изданиях.

6.1. Примерный перечень вопросов для подготовки к зачету

1. Жизненный цикл программного обеспечения. Дать краткую характеристику каждого этапа.

2. Жизненный цикл программного обеспечения. Чем занимается объектно-ориентированное программирование.
3. Стили программирования. Процедурный стиль.
4. Стили программирования. Структурный стиль.
5. Стили программирования. Объектно-ориентированный стиль.
6. Использование технологии клиент-сервер. Способы обеспечения видимости сервера клиентом.
7. Абстрагирование. Подходы классификации.
8. Методы выявления классов..
9. Критерии проверки правильности построения класса.
10. Обработка исключительных ситуаций.
11. Типы многозадачности. Преимущества и недостатки.
12. Технология COM. Понятие интерфейса.
13. COM-сервер. Определение. Типы COM-серверов.
14. Модели STA и MTA. Преимущества и недостатки каждой модели.
15. Компонент CLR.
16. Промежуточный язык MSIL.
17. Выполнение .NET программы.
18. Понятие сборки. GAC.
19. Дать определение тестированию и отладке. Особенности и объекты тестирования. Автономное и комплексное тестирование.
20. Дать определение тестированию и отладке. Направления тестирования. Стратегия тестирования. Контрольный лист тестирования модуля.
21. Дать определение тестированию и отладке. Локализация ошибок. Классификация ошибок. Безопасное программирование.
22. Оценки ошибок.

6.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов самостоятельной работы

1. Перечислите области человеческой деятельности, где используются программные продукты.
2. Что такое программная система?
3. Что такое жизненный цикл программного обеспечения?
4. Перечислите этапы жизненного цикла программного обеспечения.
5. Перечислите основные подходы программирования.
6. Опишите структурный подход программирования.
7. Опишите процедурный подход программирования.
8. Опишите объектно-ориентированный подход программирования.
9. Дайте определение процедуры и функции.
10. Дайте определения программирования.
11. Дайте определение структуры данных.
12. Как Вы понимаете понятие «интерфейс».
13. Что такое поток управления и поток данных?
14. Что такое объект? Что может быть объектом?
15. Что такое класс? Чем отличается класс от объекта?

16. Чем характеризуется класс?
17. Что такое состояние класса?
18. Что такое поведение класса?
19. Перечислите основные типы операций.
20. Что такое «отношения между классами»?
21. Что такое модель и моделирование?
22. Что такое «классификация»?
23. Что такое качество программного обеспечения с Вашей точки зрения?
24. Что такое тестирование программных продуктов?

6.3. Примерные планы практических занятий

Задание 1. Написать программу, разбирающую журнал событий (например, журнал web-сервера, syslog и т.д.) и вычисляющий ежемесячную статистику в зависимости от содержимого выбранного журнала (например, обращения с различных IP-адресов, ошибки различных типов, аутентификацию пользователей и т.д.).

Задание 2. На языке Ruby реализуйте класс, аналогичный Array, с многопоточной реализацией итераторов: «map», «any?», «all?», «select». Объясните, можно ли таким образом реализовать итератор «inject?».

Не допускается использовать циклы или рекурсию.

Вся логика работы с потоками должны быть вынесена в отдельный метод, общий для всех итераторов.

Задание 3. Представление данных в виде S-выражений.

1) Определите формат представления данных с древовидной структурой в виде S-выражений. В качестве семантической основы рекомендуется рассмотреть форматы XML и/или JSON. Допускается использование как классических S-выражений, так и расширенных, используемых в языке Clojure.

2) Разработайте язык навигации и простых запросов для заданного формата. Должны поддерживаться: относительный или абсолютный путь к узлу дерева, путь с условиями на свойства промежуточных узлов, путь с переменной вложенностью. В качестве семантической основы рекомендуется рассмотреть язык XPath для XML. Реализуйте функции поиска и модификации заданного формата посредством этого языка.

Задание 4. Объектная модель.

1) Определите для языка Clojure объектную модель. Обеспечите поддержку следующих элементов:

- класс;
- атрибут (свойство, слот);
- примесь (модуль в терминах Ruby);
- одиночное наследование классов;
- множественное наследование примесей;
- диспетчеризацию обработки сообщений относительно одного аргумента (динамический полиморфизм).

2) Определите соответствующие функции и/или макросы для работы с перечисленными выше элементами.

Задание 5. Конвейерная обработка данных.

Конвейер состоит из связанных между собой узлов. Узел характеризуется набором входных и выходных каналов, по которым могут передаваться объекты.

Узел ожидает появления определенного набора данных на своих входных каналах, после чего проводит вычисления и порождает объект на своих выходных каналах (не обязательно всех). Например, узел «сумматор» имеет два входных канала и один выходной. Получая по одному объекту с каждого входного канала он их суммирует и результат отправляет в выходной канал. При этом, если входные каналы не будут синхронизованы, то в одном из них могут «скапливаться» данные, ожидающие обработки.

Определите проблемно-специфичный язык (DSL) для работы с конвейерными схемами обработки данных, поддерживающий следующие элементы:

- узлы с набором входных и выходных каналов;
- статическую типизацию каналов;
- формирование конвейера в форме связанных узлов в соответствии с типизацией;
- запуск конвейера и обработку данных;
- параллельное исполнение отдельных узлов конвейера;
- использование одинаковых узлов для формирования разных конвейеров.

Продемонстрируйте работу разработанной вычислительной модели на примере моделирования производства автомобилей. Должны быть реализованы технологические цепочки производства элементов автомобиля и его сборки. Ряд «ресурсов» должен быть использован в нескольких цепочках, должны быть предусмотрены узлы распределители.

Кроме того, должны быть предусмотрены узлы, обеспечивающие «контроль качества», т.е. отбрасывающие бракованные детали.

Допускается придумать свою модельную задачу, согласовав ее с преподавателем. Дополнительно. Обеспечьте поддержку следующих элементов:

- настраиваемые узлы, параметры которых можно менять в уже созданном конвейере;
- поддержку конвейеров с циклами;
- признак окончания работы конвейера (обратите внимание, что конвейер может прийти в состояние, когда его работа никогда не завершится, в этом случае необходимо диагностировать ошибку);
- продемонстрируйте работу дополнительных элементов на модельной задаче;
- постройте для модельной задачи конвейер с автобалансировкой; для сборочного конвейера это означает, что если возникает узкое место по причине нехватки деталей, то узлы-распределители должны быть перенастроены соответствующим образом (не выходя за рамки вычислительной модели).

Критерии оценки уровня овладения слушателем компетенциями на этапе зачета по учебной дисциплине

Оценка	Характеристики ответа слушателя
Зачтено	слушатель показал глубокие знания программного материала, грамотно и логично его излагает, быстро принимает правильные решения, в ходе ответа демонстрирует глубокие знания основной и дополнительной литературы, умеет применять полученные знания к будущей профессиональной деятельности; если слушатель твердо знает программный материал, грамотно его излагает, не допускает существенных неточностей в ответе, правильно применяет полученные знания при решении практических вопросов, демонстрирует твердые знания основной литературы; если слушатель имеет знания только основного материала, но не усвоил его детали, не допускает грубых ошибок в ответе, требует в отдельных случаях наводящих вопросов для принятия правильного решения.
Не зачтено	слушатель не раскрыл содержание вопросов, не знает основной и дополнительной литературы в целом, отсутствуют знания по проблемам программного материала.

7. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Введение в программные системы и их разработку [Электронный ресурс]/ С.В. Назаров [и др.].— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 649 с.— Режим доступа: <http://www.iprbookshop.ru/16698>.— ЭБС «IPRbooks», по паролю.

2. Долженко А.И. Технологии командной разработки программного обеспечения информационных систем [Электронный ресурс]/ Долженко А.И.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 300 с.— Режим доступа: <http://www.iprbookshop.ru/39569>.— ЭБС «IPRbooks», по паролю.

3. Разработка программного обеспечения системы мониторинга производства на языке C++ с использованием математической модели технологического процесса [Электронный ресурс]: учебное пособие/ А.А. Хвостов [и др.].— Электрон. текстовые данные.— Воронеж: Воронежский государственный университет инженерных технологий, 2014.— 117 с.— Режим доступа: <http://www.iprbookshop.ru/47444>.— ЭБС «IPRbooks», по паролю.

4. Сычев А.В. Перспективные технологии и языки веб-разработки [Электронный ресурс]/ Сычев А.В.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 493 с.— Режим доступа: <http://www.iprbookshop.ru/39643>.— ЭБС «IPRbooks», по паролю.

б) дополнительная литература:

1. Батоврин В.К. Системная и программная инженерия. Словарь-справочник [Электронный ресурс]: учебное пособие для вузов/ Батоврин В.К.— Электрон. текстовые данные.— М.: ДМК Пресс, 2010.— 280 с.— Режим доступа: <http://www.iprbookshop.ru/7972>.— ЭБС «IPRbooks», по паролю.

2. Игнатова Е.В. Язык информационных технологий [Электронный ресурс]: учебное пособие/ Игнатова Е.В.— Электрон. текстовые данные.— М.: Евразийский открытый институт, 2011.— 75 с.— Режим доступа: <http://www.iprbookshop.ru/11143>.— ЭБС «IPRbooks», по паролю.

3. Информационные технологии. Часть 3. Сетевые технологии [Электронный ресурс]: учебно-методический комплекс по специальности 071201 «Библиотечно-информационная деятельность», квалификаций «Референт-аналитик информационных ресурсов», «Библиотекарь-библиограф, преподаватель»/ — Электрон. текстовые данные.— Кемерово: Кемеровский государственный институт культуры, 2012.— 56 с.— Режим доступа: <http://www.iprbookshop.ru/29665>.— ЭБС «IPRbooks», по паролю.

4. Информационные технологии. Часть 5. Менеджмент информационных технологий [Электронный ресурс]: учебно-методический комплекс по специальности 052700 (071201) – «Библиотечно-информационная деятельность»/ — Электрон. текстовые данные.— Кемерово: Кемеровский государственный институт культуры, 2012.— 31 с.— Режим доступа: <http://www.iprbookshop.ru/29666>.— ЭБС «IPRbooks», по паролю.

5. Куликов И.М. Технологии разработки программного обеспечения для математического моделирования физических процессов. Часть 1. Использование суперкомпьютеров, оснащенных графическими ускорителями [Электронный ресурс]: учебное пособие/ Куликов И.М.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2013.— 40 с.— Режим доступа: <http://www.iprbookshop.ru/45044>.— ЭБС «IPRbooks», по паролю.

6. Малявко А.А. Системное программное обеспечение. Формальные языки и методы трансляции. Часть 3 [Электронный ресурс]: учебное пособие/ Малявко А.А.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2012.— 120 с.— Режим доступа: <http://www.iprbookshop.ru/45019>.— ЭБС «IPRbooks», по паролю.

7. Павлова Е.А. Технологии разработки современных информационных систем на платформе Microsoft .NET [Электронный ресурс]/ Павлова Е.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 128 с.— Режим доступа: <http://www.iprbookshop.ru/16101>.— ЭБС «IPRbooks», по паролю.

8. Смирнов А.А. Прикладное программное обеспечение [Электронный ресурс]: учебное пособие/ Смирнов А.А.— Электрон. текстовые данные.— М.: Евразийский открытый институт, 2011.— 384 с.— Режим доступа: <http://www.iprbookshop.ru/11079>.— ЭБС «IPRbooks», по паролю.

9. Технологии разработки и создания компьютерных сетей на базе аппаратуры D-LINK [Электронный ресурс]: учебное пособие для вузов/ В.В. Баринов [и др.].— Электрон. текстовые данные.— М.: Горячая линия - Телеком, 2013.— 216 с.— Режим доступа: <http://www.iprbookshop.ru/37207>.— ЭБС «IPRbooks», по паролю.

10. Яковлев В.В. Технологии виртуализации и консолидации информационных ресурсов [Электронный ресурс]: учебное пособие/ Яковлев В.В.— Электрон. текстовые данные.— М.: Учебно-методический центр по образованию на железнодорожном транспорте, 2015.— 156 с.— Режим доступа: <http://www.iprbookshop.ru/45322>.— ЭБС «IPRbooks», по паролю.

в) интернет-ресурсы:

1. <http://www.compulenta.ru/> – Интернет-издание «Компьюлента», посвященное новостям компьютерной индустрии, науки и техники.
2. <http://www.wisesoft.ru/> – Каталог журналов (большой выбор компьютерных журналов).
3. <http://infl.info/> – Планета информатики.
4. <http://www.teachvideo.ru/faq> – Коллекция видеоуроков по разным сферам IT-тематики.
5. <http://www.spohelp.ru/> – Пакет свободного программного обеспечения для образовательных учреждений РФ.
6. <http://www.intuit.ru/> – Интернет университет информационных технологий.
7. <http://newb.by.ru/index.html> – Учебные языки программирования.

8. Материально-техническое обеспечение дисциплины

Учреждение располагает материально-технической базой, обеспечивающей проведение всех видов лекционных, семинарских и практических занятий.

Слушатели имеют доступ с компьютеров, входящих в локальную сеть и сеть Wi-Fi, в Интернет.

В Учреждении организованы учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Данные аудитории укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа используются слайд-лекции с обратной связью (интерактивные).

9. Методические указания слушателям по освоению дисциплины

Методические указания для слушателей по организации самостоятельной работы в процессе освоения дисциплины

Самостоятельная работа имеет целью закрепление и углубление знаний и навыков, полученных на лекциях и семинарских занятиях по дисциплине, подготовку к зачету, а также формирование культуры умственного труда и самостоятельности в поиске и приобретении новых знаний.

Основными видами самостоятельной работы являются:

- изучение отдельных теоретических вопросов при подготовке к семинарам, в том числе подготовка докладов, сообщений, рефератов по данным вопросам;
- осмысление информации, сообщаемой преподавателем, ее обобщение и краткая запись;

- своевременная доработка конспектов лекций;
- подбор, изучение, анализ и конспектирование рекомендуемой литературы;
- подготовка к зачету.