

МИНОБРНАУКИ РОССИИ

**Воткинский филиал
Федерального государственного бюджетного образовательного
учреждения высшего образования
«Ижевский государственный технический университет имени М.Т. Калашникова»
(ВФ ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»)**

УТВЕРЖДАЮ



Директор

И.А. Давыдов

«25»

июня

2018 г.

РАБОЧАЯ ПРОГРАММА

по дисциплине: Объектно-ориентированное программирование

для направления: 09.03.01 «Информатика и вычислительная техника»

профиль «Автоматизированные системы обработки информации и управления»

форма обучения: очная

программа подготовки: академический бакалавриат

общая трудоемкость дисциплины составляет: 5 зачетных единиц(ы)

Вид учебной работы	Всего часов	Семестры			
		4			
Аудиторные занятия (всего)	32	32			
В том числе:	-	-	-	-	-
Лекции	16	16			
Практические занятия (ПЗ)					
Семинары (С)					
Лабораторные работы (ЛР)	16	16			
Самостоятельная работа (всего)	148	148			
В том числе:	-	-	-	-	-
Курсовой проект (работа)	КР	КР			
Расчетно-графические работы/КТР					
Реферат					
<i>Другие виды самостоятельной работы</i>					
Вид промежуточной аттестации (зачет, экзамен)	Зачет-2	Зачет-2			
Общая трудоемкость	час	180	180		
	зач. ед.	5	5		


Кафедра «Организация вычислительных процессов и систем управления»

Составитель: Крутихин Алексей Дмитриевич, кандидат технических наук, доцент
Замятин Константин Игоревич, кандидат технических наук, доцент

Рабочая программа составлена на основании ФГОС ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника (уровень бакалавриата) №5 от 12.01.2016г. и утверждена на заседании кафедры

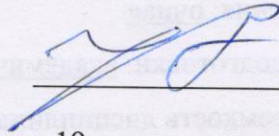
Протокол от « 19 » апреля 2018 г. № 04/18

Директор Воткинского филиала «ИжГТУ имени М.Т. Калашникова»


И.А. Давыдов
« 19 » апреля 2018 г.

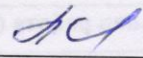
СОГЛАСОВАНО

Председатель учебно-методической комиссии
по направлению 09.03.01 «Информатика и вычислительная техника»,
профиль «Автоматизированные системы обработки
информации и управления»


К.Б. Сентяков
« 19 » апреля 2018 г.

Количество часов рабочей программы соответствует количеству часов рабочего учебного плана направления 09.03.01 «Информатика и вычислительная техника», профиль «Автоматизированные системы обработки информации и управления»

Ведущий специалист учебной части
ВФ ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»


Соловьева Л.Н.
« 19 » апреля 2018 г.

Дисциплина		Объектно-ориентированное программирование				
Номер		<i>Академический год</i>			<i>семестр</i>	
<i>кафедра</i>		<i>Программа</i>		09.03.01 «Информатика и вычислительная техника», профиль «Автоматизированные системы обработки информации и управления»		
<i>Составитель</i>		Замятин К.И., к.т.н, доцент				
<i>Цели и задачи дисциплины, основные темы</i>		<p>Цели: Ознакомление студентов с основными концепциями, составляющими основу объектно-ориентированного программирования, изучение основ объектно-ориентированного программирования.</p> <p>Задачи: Овладение основными навыками построения объектно-ориентированной программы,</p> <p>Знания: Основные постулаты объектно-ориентированного программирования, способы определения и использования классов, взаимосвязь экземпляров класса, принцип построения объектно-ориентированной программы, концепцию иерархии классов.</p> <p>Умения: Создавать программы с регулируемым доступом к элементам класса, самостоятельно осваивать новые возможности сред объектно-ориентированного программирования.</p> <p>Навыки: Владеть навыком создания индивидуальных и групповых проектов с помощью объектно-ориентированного метода программирования.</p> <p>Лекции (основные темы): Основные постулаты ООП. Понятие класса объекта. Конструкторы и деструкторы. Перегрузка операций класса. Наследование классов и полиморфизм ООП.</p> <p>Лабораторные работы: Работа с MS Visual C++.</p>				
<i>Основная литература</i>		<p>Лисицин, Д. В. Объектно-ориентированное программирование [Электронный ресурс] : конспект лекций / Д. В. Лисицин. — Электрон. текстовые данные. — Новосибирск : Новосибирский государственный технический университет, 2010. — 88 с. — 978-5-7782-1454-5. — Режим доступа: http://www.iprbookshop.ru/44970.html</p> <p>Николаев, Е. И. Объектно-ориентированное программирование [Электронный ресурс] : учебное пособие / Е. И. Николаев. — Электрон. текстовые данные. — Ставрополь : Северо-Кавказский федеральный университет, 2015. — 225 с. — 2227-8397. — Режим доступа: http://www.iprbookshop.ru/62967.html</p>				
<i>Технические средства</i>		Лаборатория информационных технологий. Компьютерный класс и учебная аудитория для проведения занятий лекционного и семинарского типа, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации, групповых и индивидуальных консультаций.				
<i>Компетенции</i>		Приобретаются студентами при освоении модуля				
		ОПК-1 способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем ОПК-4 способностью участвовать в настройке и наладке программно-аппаратных комплексов ПК-2 способностью разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования				
<i>Зачетных единиц</i>	5	<i>Форма проведения занятий</i>	<i>Лекции</i>	<i>Практические занятия</i>	<i>ЛАБОРАТОРНЫЕ РАБОТЫ</i>	<i>Самостоятельная работа</i>
		<i>Всего часов</i>	16		16	148
<i>Виды контроля</i>	<i>Диф.зач /зач/ экз</i>	<i>КП/КР</i>	<i>Условие зачета дисциплины</i>	Получение оценки «ЗАЧЕНО» и оценки 3,4,5 за КР	<i>Форма проведения самостоятельной работы</i>	Подготовка к лабораторным занятиям и зачету, КР
<i>Перечень дисциплин, знание которых необходимо для изучения данной дисциплины</i>			«Информатика», «Программирование».			

1. Цели и задачи дисциплины:

Целью преподавания дисциплины является ознакомление студентов с основными концепциями, составляющими основу объектно-ориентированного программирования, изучение основ объектно-ориентированного программирования.

Задачи дисциплины:

- теоретическое изучение основных принципов объектного подхода
- приобретение практических навыков разработки объектно-ориентированных программ

В результате изучения дисциплины студент должен

знать:

- основные постулаты объектно-ориентированного программирования
- способы определения и использования классов
- взаимосвязь экземпляров класса
- принцип построения объектно-ориентированной программы
- концепцию иерархии классов.

уметь:

- создавать программы с регулируемым доступом к элементам класса
- самостоятельно осваивать новые возможности сред объектно-ориентированного программирования.

владеть:

- основами структурного программирования в рамках одного из языков высокого уровня
- навыками разработки программ на языках высокого уровня с использованием одного из современных средств быстрой разработки программ

2. Место дисциплины в структуре ООП:

Дисциплина относится к дисциплинам по выбору Блока 1 «Дисциплины (модули)» ООП. Для изучения дисциплины студент должен

знать:

- основные принципы построения объектных моделей
- правила описания классов
- основные способы взаимодействия объектов и классов
- назначение и основы реализации виртуальных методов
- особенности и использование полиморфных объектных переменных
- особенности интерфейсных и обобщенных классов
- механизмы объектной обработки особых ситуаций
- основы построения компонентных моделей

уметь:

- описывать классы на различных объектных языках
- организовывать взаимодействие объектов и классов на основе принципов композиции и наследования
- строить иерархии классов с использованием виртуальных методов
- реализовывать контейнерные классы на основе полиморфных указателей
- описывать и использовать интерфейсные классы
- выполнять объектную обработку исключений
- создавать и использовать компонентные классы

владеть:

- навыками создания программ на языках высокого уровня

Изучение дисциплины базируется на знаниях, полученных при изучении дисциплин: Программирование, Информатика, Математика (среднее (полное) общее образование)

3. Требования к результатам освоения дисциплины:

3.1. Знания, приобретаемые в ходе изучения дисциплины

№ п/п З	Знания
1.	Постулаты объектно-ориентированного программирования
2.	Способы определения и использования классов
3.	Концепцию иерархии классов

3.2. Умения, приобретаемые в ходе изучения дисциплины

№ п/п У	Умения
1.	Создавать программы с регулируемым доступом к элементам класса
2.	Самостоятельно осваивать новые возможности сред объектно-ориентированного программирования.

3.3. Навыки, приобретаемые в ходе изучения дисциплины

№ п/п Н	Навыки
1.	Разработка программ на языках высокого уровня с использованием одного из современных средств быстрой разработки программ

3.4. Компетенции, приобретаемые в ходе изучения дисциплины

Компетенции	Знания (№№ из 3.1)	Умения (№№ из 3.2)	Навыки (№№ из 3.3)
ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	1,2,3	1,2	1
ОПК-4 способностью участвовать в настройке и наладке программно-аппаратных комплексов	1,2,3	1,2	1
ПК-2 способностью разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования	1,2,3	1,2	1

4. Структура и содержание дисциплины (модуля)

4.1. Разделы дисциплин и виды занятий

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				лек	прак	лаб	СРС*	
1	Основные понятия об объектах и классах	3	1 2 3	2		2	26	Проверочная работа
2	Взаимодействие объектов и классов	3	4 5 6	4		6	30	Выполнение лабораторных работ.
3	Принцип полиморфизма	3	7 8 9 10	2		8	30	Выполнение лабораторных работ. Контрольная работа
4	Дополнительные возможности объектных языков	3	11 12 13	4	-	-	30	Выполнение лабораторных работ.
5	Развитие объектной технологии	3	14 15 16	2	-	-	30	Выполнение лабораторных работ.
6	Подготовка к зачету	3					2	
	Всего			16		16	148	

*включая курсовое проектирование

4.2. Содержание разделов курса

№ п/п	Раздел дисциплины	Знания (номер из 3.1)	Умения (номер из 3.2)	Навыки (номер из 3.3)
1	1. История развития объектной технологии 2. Понятие об объектах 3. Классы и их описание 4. Создание и использование объектов	1	1	1
2	1. Взаимодействие на основе принципа композиции 2. Взаимодействие классов на основе наследования	3	1	1
3	1. Переопределение методов. Виртуальные методы 2. Полиморфные объектные переменные 3. Контейнерные классы	2	1	1
4	1. Интерфейсные классы 2. Обобщенные классы 3. Объектная обработка исключений	2,3	1	1
5	1. Компонентная модель Delphi 2. Компонентная модель Java Beans 3. Компонентная модель платформы .NET	1,2,3	2	1

4.3. Наименование тем лабораторных работ, их содержание и объем в часах

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудоемкость (час)
1.	1	Классы, свойства, индексаторы. Одномерные, прямоугольные и ступенчатые массивы	2
2.	2	Наследование. Исключения. Интерфейсы. Итераторы и блоки итераторов	2
3.	2	Универсальные типы. Классы-коллекции. Методы расширения класса System.Linq.Enumerable	2
4.	3	Делегаты. События	2
5	3	Классы для работы с файлами. Сериализация	4
6	3	Сериализация. Взаимодействие управляемого и неуправляемого кода	4
	Всего		16

5. Содержание самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

5.1. Содержание самостоятельной работы

№ п/п	№ раздела дисциплины	Наименование тем	Трудоемкость (час)
1.		История развития объектной технологии	8
2.		Понятие об объектах	8
3.		Классы и их описание	8
4.		Создание и использование объектов	8
5.		Взаимодействие на основе принципа композиции	8
6.		Взаимодействие классов на основе наследования	8
7.		Переопределение методов. Виртуальные методы	6

8.		Полиморфные объектные переменные	6
9.		Контейнерные классы	6
10.		Интерфейсные классы	6
11.		Обобщенные классы	6
12.		Объектная обработка исключений	8
13.		Компонентная модель Borland Delphi	8
14.		Компонентная модель Java Beans	8
15.		Компонентная модель платформы .NET	8
16.		Курсовая работа и зачёт	38
	Всего		148

5.2. Оценочные средства, используемые для текущего контроля успеваемости и промежуточной аттестации обучающихся по итогам освоения дисциплины, их виды и формы, требования к ним и шкалы оценивания приведены в приложении к рабочей программе дисциплины «Фонд оценочных средств по дисциплине «Объектно-ориентированное программирование», которое оформляется в виде отдельного документа.

6. Учебно-методическое и информационное обеспечение дисциплины:

а) Основная литература

№ п/п	Наименование книги	Год издания
1	Лисицин, Д. В. Объектно-ориентированное программирование [Электронный ресурс] : конспект лекций / Д. В. Лисицин. — Электрон. текстовые данные. — Новосибирск : Новосибирский государственный технический университет, 2010. — 88 с. — 978-5-7782-1454-5. — Режим доступа: http://www.iprbookshop.ru/44970.html	2016
2	Николаев, Е. И. Объектно-ориентированное программирование [Электронный ресурс] : учебное пособие / Е. И. Николаев. — Электрон. текстовые данные. — Ставрополь : Северо-Кавказский федеральный университет, 2015. — 225 с. — 2227-8397. — Режим доступа: http://www.iprbookshop.ru/62967.html	2018

б) Дополнительная литература

№ п/п	Наименование книги	Год издания
1	Казанский, А. А. Объектно-ориентированное программирование на языке Microsoft Visual C# в среде разработки Microsoft Visual Studio 2008 и .NET Framework. 4.3 [Электронный ресурс] : учебное пособие и практикум / А. А. Казанский. — Электрон. текстовые данные. — М. : Московский государственный строительный университет, ЭБС АСВ, 2011. — 180 с. — 2227-8397. — Режим доступа: http://www.iprbookshop.ru/19258.html	2016

г) Программное обеспечение:

1. Microsoft Office 2016.
2. Apache OpenOffice (свободно распространяемое ПО).
3. Microsoft Visual Studio Community 2017

д) методические указания:

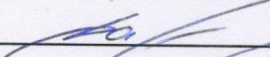
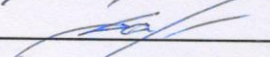
1. Николаев, Е. И. Объектно-ориентированное программирование. Часть 1 [Электронный ресурс] : лабораторный практикум / Е. И. Николаев. — Электрон. текстовые данные. — Ставрополь : Северо-Кавказский федеральный университет, 2015. — 183 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/62966.html>
2. Николаев, Е. И. Объектно-ориентированное программирование. Часть 2 [Электронный ресурс] : лабораторный практикум / Е. И. Николаев. — Электрон. текстовые данные. — Ставрополь : Северо-Кавказский федеральный университет, 2015. — 156 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/63218.html>
3. Новиков, П. В. Объектно-ориентированное программирование [Электронный ресурс] : учебно-методическое пособие к лабораторным работам / П. В. Новиков. — Электрон. текстовые данные. — Саратов : Вузовское образование, 2017. — 124 с. — 978-5-4487-0011-8. — Режим доступа: <http://www.iprbookshop.ru/64650.html>
4. Лексин А.Ю. Методические указания к курсовой работе по дисциплине «Объектно-ориентированное программирование». – Владимир, Владимирский государственный университет им. А.Г. и Н.Г. Столетовых, Кафедра физики и прикладной математики, 2012

7. Материально-техническое обеспечение дисциплины:

№№ п/п	Наименование оборудованных учебных кабинетов, объектов для проведения занятий с перечнем основного оборудования
1	Лаборатория информационных технологий. Компьютерный класс и учебная аудитория для проведения занятий лекционного и семинарского типа, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации, групповых и индивидуальных консультаций. Парты, стол преподавателя, доска аудиторная. Компьютеры - 13 шт.
2	Читальный зал. Помещение для самостоятельной работы. Центр коллективного пользования. Парты. Компьютеры - 5 шт.

Лист утверждения рабочей программы дисциплины на учебный год

Рабочая программа дисциплины утверждена на ведение учебного процесса в учебном году:

Учебный год	«Согласовано»: заведующий кафедрой, ответственной за РПД (подпись и дата)
2018-2019	 01.09.2018
2019- 2020	 01.09.2019
2020- 2021	
2021 – 2022	
2022 - 2023	
2023 - 2024	
2024- 2025	

МИНОБРНАУКИ РОССИИ

Воткинский филиал
федерального государственного бюджетного образовательного
учреждения высшего образования
«Ижевский государственный технический университет
имени М.Т. Калашникова»
(ВФ ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»)

Кафедра Организация вычислительных процессов и систем управления
(наименование кафедры)

УТВЕРЖДЕН

на заседании кафедры

«19» сентября 2018 г., протокол № 04/18

Директор филиала

 Давыдов И.А.
(подпись)

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ

Объектно-ориентированное программирование
(наименование дисциплины)

09.03.01 – Информатика и вычислительная техника
(шифр и наименование направления/специальности наименование дисциплины)

Автоматизированные системы обработки информации и управления
(наименование профиля/специализации/магистерской программы)

Бакалавр

Квалификация (степень) выпускника

Воткинск 2018

Содержание

Раздел	Стр.
Содержание	2
Паспорт фонда оценочных средств по дисциплине «Объектно- ориентированное программирование»	3
1. Зачетно-экзаменационные материалы	3
2. Комплекты оценочных средств	6
3. Темы для самостоятельной работы	8
4. Критерии формирования оценок на зачете	8

**Паспорт
фонда оценочных средств
по дисциплине**

Объектно-ориентированное программирование
(наименование дисциплины)

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	ОСНОВНЫЕ ПОНЯТИЯ ОБ ОБЪЕКТАХ И КЛАССАХ	ОПК-1 ОПК-4 ПК-2	Собеседование по вопросам лекционного материала
2	ВЗАИМОДЕЙСТВИЕ ОБЪЕКТОВ И КЛАССОВ	ОПК-1 ОПК-4 ПК-2	Собеседование по вопросам лекционного материала
3	ПРИНЦИП ПОЛИМОРФИЗМА	ОПК-1 ОПК-4 ПК-2	Собеседование по вопросам лекционного материала
4	ДОПОЛНИТЕЛЬНЫЕ ВОЗМОЖНОСТИ ОБЪЕКТНЫХ ЯЗЫКОВ	ОПК-1 ОПК-4 ПК-2	Собеседование по вопросам лекционного материала
5	РАЗВИТИЕ ОБЪЕКТНОЙ ТЕХНОЛОГИИ	ОПК-1 ОПК-4 ПК-2	Собеседование по вопросам лекционного материала

- Наименование темы (раздела) или тем (разделов) взяты из рабочей программы дисциплины.

1. Зачетно-экзаменационные материалы

Перечень контрольных вопросов для проверки остаточных знаний и для проведения экзамена.

1. Какие основные этапы можно выделить в развитии технологии программирования?
2. Какие библиотеки классов используются для создания Windows-приложений?
3. Какие события можно выделить как ключевые для развития объектной технологии в середине 90-х годов?
4. В чем заключаются основные особенности языка Java?
5. Какая технология от Microsoft является определяющей в настоящее время и в чем ее особенности?
6. Что такое объект в широком и узком смысле слова?
7. Приведите примеры объектов.
8. Что такое свойства объекта и какого они могут быть типа?
9. Что такое методы и как они классифицируются?
10. Что такое принцип инкапсуляции?
11. Какую работу выполняют конструкторы объектов?
12. Что такое методы доступа к свойствам объекта и для чего они вводятся?
13. Что такое класс и как это понятие связано с объектом?
14. Что включает в себя описание класса?
15. Что такое реальные и абстрактные классы?
16. Какие правила описания классов существуют в языке Delphi Pascal?

17. Что такое перегрузка методов и какие при этом используются правила?
18. Приведите пример описания класса на языке Delphi Pascal.
19. Какие правила описания классов существуют в языке C++?
20. Для чего необходимы директивы ограничения доступа и как они используются?
21. Какие правила именования конструкторов используются в языке C++?
22. Приведите пример описания класса на языке C++.
22. Какие правила описания классов существуют в языке Java?
23. Приведите пример описания класса на языке Java.
24. Какие способы создания объектов существуют в разных языках?
25. Как создаются статические объекты в языке C++?
26. Как объявляются и создаются динамические объекты в языке C++?
27. Как объявляются и создаются объекты в языке Java?
28. Как объявляются и создаются объекты в языке Delphi Pascal?
29. Как выполняется вызов методов в разных языках?
30. Какие способы взаимодействия классов являются основными?
31. В каких случаях между классами можно установить отношение агрегации?
32. Какие возможности дает использование механизма агрегации классов?
33. Приведите примеры агрегации и композиции объектов.
34. Какие шаги надо выполнить для программной реализации отношения агрегации?
35. Как необходимо строить конструкторы классов, взаимодействующих на основе агрегации?
36. Приведите примеры программной реализации композиции объектов.
37. Как графически задается отношение агрегации между классами?
38. В чем состоит отличие отношений агрегации и композиции?
39. В каких случаях между классами можно установить отношение наследования?
40. В чем состоит суть механизма наследования?
41. В чем состоят отличия агрегации и наследования?
42. Чем дочерние подклассы отличаются от своих родителей?
43. Что такое иерархия классов?
44. Чем множественное наследование отличается от простого?
45. В чем преимущества и недостатки множественного наследования?
46. Какие преимущества дает использование механизма наследования?
47. Как задается отношение наследования в разных языках?
48. Что происходит при отсутствии родительского класса в заголовке нового класса?
49. Приведите примеры иерархий классов.
50. Как изменяется механизм ограничения доступа к элементам класса при использовании наследования?
51. Какие правила надо соблюдать при реализации конструкторов дочерних классов?
52. Как вызываются родительские конструкторы в разных языках?
53. Что такое полиморфизм и какие проявления он имеет?
54. Что такое переопределение методов и чем оно отличается от перегрузки?
55. Чем объясняется необходимость использования механизма переопределения методов?
56. Что такое статическая и динамическая компоновка, их преимущества и недостатки?
57. Что такое таблица виртуальных методов и для чего она используется?
58. Какие особенности имеет иерархия графических фигур при использовании виртуальных методов?
59. Какие связи выстраиваются в памяти при выполнении объектной программы с виртуальными методами?
60. Какие особенности имеет использование механизма виртуальных методов в разных языках?
61. Что такое полиморфные объектные указатели?

62. Какие правила используются в механизме полиморфных указателей?
63. Какие преимущества дает объединение полиморфных указателей в массив?
64. Как используются полиморфные указатели при передаче параметров методам?
65. Какие сложности возникают при использовании полиморфных указателей?
66. Что такое метаданные и что они содержат?
67. Какие связи между таблицами-структурами создаются при выполнении объектной программы?
68. Как выполняется проверка текущего динамического типа полиморфных указателей?
69. Как можно организовать выборочную обработку объектов некоторой подиерархии?
70. Как и зачем выполняется приведение типов полиморфных указателей?
71. Что выполняют операторы `is` и `as` в языке Delphi Pascal?
72. Как выполняется проверка динамического типа объектного указателя и его приведение в языке Java?
73. Что такое контейнеры и как их можно реализовать?
74. Как реализуется контейнерный класс на основе обычного массива?
75. Как реализуется контейнерный класс на основе динамического массива?
76. Как реализуется контейнерный класс на основе динамического списка?
77. Какие классы называются интерфейсными?
78. С какими целями вводятся интерфейсные классы?
79. Какой механизм наследования доступен для интерфейсных классов?
80. Как взаимодействуют интерфейсные и реальные классы?
81. Какие типы классов может включать объектная модель приложения?
82. Чем интерфейсные классы отличаются от абстрактных классов?
83. Как выполняется описание интерфейсных классов в языке Java?
84. Как принцип полиморфизма применяется к интерфейсным типам?
85. Какие интерфейсы являются базовыми для стандартной библиотеки контейнеров языка Java?
86. Какие контейнеры реализованы в стандартной библиотеке классов языка Java?
87. Что такое итераторы и как они используются?
88. Какие методы объявляются в стандартных итераторах языка Java?
89. Что такое компараторы и как они используются?
90. Как объявляются интерфейсные классы в языке Delphi Pascal?
91. Что такое обобщенные классы?
92. Как объявляются шаблонные функции в языке C++?
93. Когда и каким образом обрабатываются шаблонные функции?
94. Приведите пример описания и использования шаблонной функции.
95. Что такое шаблонный класс и что он может содержать?
96. Приведите пример описания шаблонного класса.
97. Как выполняется описание методов шаблонного класса за пределами класса?
98. Что происходит при обработке объявлений объектов шаблонных классов?
99. Что такое STL?
100. Что такое исключения?
101. В чем смысл объектной обработки исключений?
102. Приведите примеры наиболее типичных классов исключений.
103. Что необходимо сделать в программах для обработки исключений?
104. Приведите общую структуру программного кода для обработки исключений?
105. Как оформляется блок завершения и чем он отличается от блока обработки исключений?
106. Как программно можно сгенерировать исключение?
107. Что такое компонентные классы и для чего они вводятся?
108. Что такое свойства (`property`) в компонентной модели и как они описываются?

109. Приведите пример описания класса, использующего понятие свойства.
110. Как свойства используются в программах?
111. Какие свойства называются опубликованными?
112. Как в компонентной модели Borland Delphi реализуется обработка событий?
113. Как в классах объявляются свойства процедурного типа?
114. Какие стандартные процедурные типы объявлены в библиотеке VCL?
115. Как можно реализовать обработку нестандартных событий?
116. Какие базовые классы в библиотеке VCL поддерживают компонентную модель?
117. Что такое свойство потоковости и для чего оно необходимо?
118. Что обеспечивает механизм владения и как он реализуется?
119. Что такое визуальные и невидимые компоненты и как они используются?
120. Перечислите основные концепции компонентной модели Java Beans.
121. Какие правила используются для описания компонентных классов в модели Java Beans?
122. Приведите пример описания компонентного класса Java Beans.
123. Как реализуется обработка событий в компонентной модели Java Beans?
124. Назовите основные классы событий в модели Java Beans.
125. Перечислите основные интерфейсы слушателя в модели Java Beans.
126. Как создаются и регистрируются объекты-слушатели в модели Java Beans?
127. Что обеспечивает механизм сериализации в языке Java?
128. Как объявляются свойства в языке C# ?
129. Что такое делегаты и для чего они вводятся?
130. Как описываются и используются делегаты?
131. Как можно пополнить список вызовов делегата?
132. Как в языке C# вводится обработка событий?
133. В чем состоят особенности создания распределенных приложений?
134. Какие технологии используются при создании распределенных приложений?
135. В чем состоят особенности технологии MS COM/DCOM?
136. Что лежит в основе создания распределенных приложений для платформы Java?
137. Что такое технология CORBA?
138. Что такое SOAP?
139. Для чего используется язык UML?
140. Что такое паттерны проектирования?

2. Комплекты оценочных средств

2.1. Вопросы к собеседованию по лекционному материалу на темы

1. Программа в объектно-ориентированном программировании состоит из:
 - стандартных функций
 - событийных процедур
 - переменных
 - блок-схем
2. Доступ к изменению свойств объекта возможен только через принадлежащие ему ...
 - события
 - методы
 - подмножества
 - классы
3. Является примером события в объектно-ориентированной среде(выберите несколько ответов):

- движение мыши
 - нажатие клавиши на клавиатуре
 - щелчок кнопкой мыши
 - двойной щелчок кнопкой мыши
4. Примеров, в котором некоторое свойство объекта получает новое значение, является(выберите несколько ответов):
- Form1.Close
 - Form1.Color=Blue
 - Label1.Caption="Введите число"
 - Documents(«1.doc»).PrintOut
5. Не является основным понятием объектно-ориентированного программирования:
- объект
 - свойство
 - метод
 - событие
 - движение
6. Является методом стандартного окна Windows(выберите несколько ответов):
- свернуть
 - распечатать
 - вырезать
 - закрыть
 - помыть
7. В момент времени объект характеризуется:
- событиями
 - переменными
 - методами
 - свойствами
8. В записи Documents(«1.doc»).PrintOut методом является:
- Document
 - 1.doc
 - Print
 - Out
 - PrintOut
9. В записи Form1.Color = Red объектом является:
- Form1
 - Color
 - Red
 - нет объекта
10. Является свойством объекта "файл"(выберите несколько ответов):
- размер файла
 - копировать
 - удалить
 - имя

3. Темы для самостоятельной работы

Варианты заданий для самостоятельной работы: поиск учебных пособий по данному материалу, подготовка презентации и доклада

1. Основные понятия об объектах и классах
2. Взаимодействие объектов и классов
3. Принцип полиморфизма
4. Дополнительные возможности объектных языков
5. Развитие объектной технологии

Материалы по курсовому проектированию сформулированы в методических указаниях.

4. Критерии формирования оценок на зачете

На собеседовании задается два вопроса. Критерии формирования оценок по результатам собеседования:

- «незачтено» - обучающийся не ответил правильно ни на один вопрос;
- «зачтено» - а) обучающийся развернуто и правильно ответил на один вопрос.
- б) обучающийся развернуто и правильно ответил на два вопроса.