

МИНОБРНАУКИ РОССИИ
Ярославский государственный университет им. П.Г. Демидова

Кафедра вычислительных и программных систем

УТВЕРЖДАЮ
Декан факультета ИВТ
 Д.Ю. Чалый
« 23 » мая 2023 г.

Рабочая программа дисциплины
«Объектно-ориентированное программирование»

Направление подготовки
02.03.02 Фундаментальная информатика и информационные технологии

Профиль
«Информатика и компьютерные науки»

Квалификация выпускника
Бакалавр

Форма обучения
очная

Программа рассмотрена
на заседании кафедры
от 21 апреля 2023 г.,
протокол № 8

Программа одобрена НМК
факультета ИВТ
протокол № 6 от
28 апреля 2023 г.

Ярославль

1. Цели освоения дисциплины

Целями дисциплины «Объектно-ориентированное программирование» являются изучение основных этапов, методов, средств и стандартов разработки программного обеспечения, детальное изучение одного из объектно-ориентированных языков программирования (Java), обучение студентов разработке кросс-платформенных приложений.

2. Место дисциплины в структуре ОП бакалавриата

Дисциплина «Объектно-ориентированное программирование» относится к вариативной части ОП бакалавриата.

Для освоения данной дисциплиной студенты должны обладать знаниями по математике и информатике в объеме школьной программы, а так же в объеме курса «Основы программирования», проявлять настойчивость, целеустремленность и инициативу в процессе обучения.

Знания и навыки, полученные при ее изучении, используются учащимися при изучении последующих дисциплин профессиональной и вариативной частей ОП, а именно «Программная инженерия», «Современные мобильные платформы и сервисы», а также для продолжения обучения в магистратуре по направлению Фундаментальная информатика и информационные технологии. Материал лекций является основой для двух компьютерных практикумов: «Практикум ЭВМ по объектно-ориентированному программированию» и «Практикум ЭВМ по языкам программирования».

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП бакалавриата

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Формируемая компетенция (код и формулировка)	Индикатор достижения компетенции (код и формулировка)	Перечень планируемых результатов обучения
Общепрофессиональные компетенции		
ОПК-2 Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	ОПК-2.1 Обладает навыками выбора методов и моделей для решения прикладных задач в профессиональной деятельности ОПК-2.2 Разрабатывает и реализует алгоритмы решения прикладных задач с использованием современных систем программирования	Знать: – основы теории алгоритмов и ее применения; – синтаксис языка программирования Java; – принципы объектно-ориентированного программирования. Уметь: – разрабатывать программы на языке Java; – создавать консольные пользовательские приложения с использованием JavaFX; – выполнять отладку программы. Владеть навыками: – работы с интегрированной средой

		разработки NetBeans; – работы с научно-технической литературой и технической документацией Java.
--	--	---

4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зач. ед., 108 акад. час.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)					Формы текущего контроля успеваемости		Форма промежуточной аттестации (по семестрам)
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа		
			Контактная работа							
1.	Объектно-ориентированное программирование. Основные понятия.	3	4						7	
2.	Синтаксис языка Java	3	2			1			7	Задания для самостоятельной работы
3.	Классы. Элементы класса.	3	4						8	Контрольная работа №1
4.	Управление доступом. Инкапсуляция.	3	4						8	Лабораторная работа
5.	Наследование и полиморфизм	3	4			1			8,7	Задания для самостоятельной работы
6.	Интерфейсы	3	5						8	Лабораторная работа
7.	Обработка исключительных ситуаций.	3	5						8	Лабораторная работа
8.	Основные классы для ввода/вывода.	3	4						7	Контрольная работа №2
9.	Обработка текстовой информации с использованием класса String.	3	4			1			7	Задания для самостоятельной работы
	Всего за 3 семестр		36			3			68,7	Зачет
	Всего		36			3			68,7	

Содержание разделов дисциплины:

1. Объектно-ориентированное программирование. Основные понятия. Объект, класс, метод. Инкапсуляция. Наследование. Полиморфизм.
2. Синтаксис языка Java. Базовые конструкции языка Java. Переменные. Типы данных, преобразование типов.
3. Классы. Элементы класса. Поля и методы. Конструкторы. Пример разработки пользовательского класса Point.
4. Управление доступом. Инкапсуляция. Доступ к членам класса.
5. Наследование и полиморфизм. Базовые и производные классы. Доступ к элементам класса. Переопределение методов. Абстрактные классы. Примеры
6. Интерфейсы. Определение интерфейса. Реализация интерфейса. Стандартные интерфейсы.
7. Обработка исключительных ситуаций. Синтаксис, ключевые слова. Типы исключений. Общая схема работы с исключительными ситуациями: генерация исключений, перехват исключений, блоки-ловушки.
8. Основные классы для ввода/вывода. Иерархия классов. Работа с файлами.
9. Обработка текстовой информации с использованием класса String. Основные методы класса, операции со строками.

5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

Вводная лекция – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

Академическая лекция (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Лекция-беседа или «диалог с аудиторией», является наиболее распространенной и сравнительно простой формой активного вовлечения студентов в учебный процесс. Эта лекция предполагает непосредственный контакт преподавателя с аудиторией. Преимущество лекции-беседы состоит в том, что она позволяет привлекать внимание студентов к наиболее важным вопросам темы, определять содержание и темп изложения учебного материала с учетом особенностей студентов.

Мастер-класс – это особая форма учебного занятия, когда преподаватель-мастер передает свой опыт путем прямого и комментированного показа последовательности действий, методов, приемов и форм педагогической деятельности. Целью проведения мастер-класса является профессиональное, интеллектуальное и эстетическое воспитание студентов, и прежде всего, развитие в ходе мастер-класса способности студента самостоятельно и нестандартно мыслить.

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного

программного обеспечения и информационных справочных систем (при необходимости)

В процессе осуществления образовательного процесса используются: для разработки документов, презентаций, для работы с электронными таблицами

OfficeStd 2013 RUS OLP NL Acdmc 021-10232

LibreOffice (свободное)

издательская система LaTeX;

– для выполнения лабораторных работ

среда разработки NetBeans 8

(свободная)

– для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next").

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная:

1. Васильев А. Н. Java: объектно-ориентированное программирование для магистров и бакалавров.: базовый курс по объектно-ориентированному программированию. / А. Н. Васильев - СПб.: Питер, 2014. - 396 с.

2. Лагутина, Н. С., Разработка программных приложений : практикум для студентов, обучающихся по направлению Фундаментальная информатика и информационные технологии / Н. С. Лагутина, Ю. А. Ларина, А. М. Васильев; Яросл. гос. ун-т., Ярославль, ЯрГУ, 2014, 71 с.

3. Лагутина, Н. С., Разработка программных приложений [Электронный ресурс] : практикум для студентов, обучающихся по направлению Фундаментальная информатика и информационные технологии / Н. С. Лагутина, Ю. А. Ларина, А. М. Васильев; Яросл. гос. ун-т., Ярославль, ЯрГУ, 2014, 71 с. <http://www.lib.uniyar.ac.ru/edocs/iuni/20140402.pdf>

б) дополнительная:

1. Парамонов, И. В., Язык программирования Java и Java - технологии : учеб. пособие / И. В. Парамонов ; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2006, 91с

2. Парамонов, И. В., Язык программирования Java и Java - технологии [Электронный ресурс] : учеб. пособие / И. В. Парамонов ; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2006, 91с. <http://www.lib.uniyar.ac.ru/edocs/iuni/20060480.pdf>

3. Гонсалвес, Э., Изучаем Java EE 7 / Э. Гонсалвес; [пер. с англ. Е. Зазноба и др.], СПб., Питер, 2016, 640с

4. Эккель, Б., Философия Java / Б. Эккель; пер. с англ. Е. Матвеева. - 4-е полное изд., М., Питер, 2017, 1165с

в) ресурсы сети «Интернет»

– Среда разработки NetBeans 8.0.3: www.netbeans.org. Доступ свободный

– Документация java 8: <https://docs.oracle.com/javase/8/docs/api/>. Доступ свободный

– Документация javaFX: <https://docs.oracle.com/javase/8/javafx/api/toc.htm>. Доступ свободный

8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

-учебные аудитории для проведения занятий лекционного типа и практических занятий (семинаров);

-учебные аудитории для проведения лабораторных занятий;

- учебные аудитории для проведения групповых и индивидуальных консультаций,

- учебные аудитории для проведения текущего контроля и промежуточной аттестации;

-помещения для самостоятельной работы;

-помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, хранящиеся на электронных носителях и обеспечивающие тематические иллюстрации, соответствующие рабочим программам дисциплин.

Помещения для лабораторных занятий и самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

Число посадочных мест в лекционной аудитории больше либо равно списочному составу потока, а в аудитории для практических занятий (семинаров) – списочному составу группы обучающихся.

Автор(ы) :

Доцент кафедры ВПС, к.ф.-м.н. _____ Н.С.Лагутина

**Приложение №1 к рабочей программе дисциплины
«Объектно-ориентированное программирование»
Фонд оценочных средств
для проведения текущей и промежуточной аттестации студентов
по дисциплине**

**1. Типовые контрольные задания или иные материалы, необходимые для
оценки знаний, умений, навыков и (или) опыта деятельности,
характеризующих этапы
формирования компетенций**

**1.1. Контрольные задания и иные материалы, используемые в процессе
текущей аттестации**

Задания для самостоятельной работы

Примеры заданий по теме «Синтаксис языка Java»:

1. Вводится натуральное число N. Представьте его в виде суммы степеней двойки.
Результат напечатать.

Возможные тесты

Входные данные	Результат
5	1+4
255	1+2+4+8+16+32+64+128
1	1
0	Введено не натуральное число

2. В массиве целых чисел найти наиболее часто встречающееся число. Если таких чисел несколько, то определить наименьшее из них.

Возможные тесты

Входные данные	Результат
1 2 3 4	1
2 5 78 13 2 5 78 2 5 6	2
2 1 2	2
6	6

Критерии оценивания

Оценка	Критерии
Отлично	Знает и применяет подходящие для решения алгоритмы, выбирает наиболее эффективный алгоритм. Знает и применяет синтаксические конструкции языка Java. Программа выполняется для всех тестов. Создает полноценное консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости
Хорошо	Знает и применяет подходящие для решения стандартные алгоритмы. Знает и применяет синтаксические конструкции языка Java. Программа выполняется для всех тестов, если какой-либо из тестов не выполняется программа дорабатывается. Создает консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости с небольшими неточностями
Удовлетворительно	Знает и применяет простые алгоритмы. Знает и применяет синтаксические конструкции языка Java. Программа выполняется для всех тестов, возможно кроме одного-двух крайних случаев. Создает консольное приложение в среде разработки. С трудом

	поясняет код, не может изменить код при усложнении или существенном дополнении задачи
Неудовлетворительно	Не может подобрать подходящие для решения алгоритмы. Ошибается во многих синтаксических конструкциях языка Java. Программа не выполняется для большинства тестов. Не может создать консольное приложение в среде разработки или создает неработающий проект. Не может пояснить код и изменить его.

Примеры заданий по теме «Наследование и полиморфизм»:

1. Опишите модель данных для вычисления заработной платы работников организации: работающих постоянно (их заработная плата определяется размером должностного оклада и дополнительной премией в процентах от оклада) и работающих по договору (их заработная плата определяется суммой денег, выплачиваемых за один день работы и количеством отработанных дней). Необходимо вывести на экран зарплату инженера с окладом 20 000 рублей и премией 40 %, а также дизайнера, проработавшего 12 дней с оплатой 950 рублей в день. Разработать модель данных, основанную на иерархии классов. Базовый класс описывает общие поля и методы работников, в классах наследниках добавляются собственные свойства и описывается поведение.

2. Опишите модель данных для вычисления площадей полной поверхности и объемов геометрических фигур в пространстве: конусов ($S = \pi R^2 + \pi R L$, $V = 1/3 \pi R^2 H$) и прямоугольных параллелепипедов. Необходимо вывести на экран площадь полной поверхности и объем конуса с радиусом основания 3 и высотой 4, а также прямоугольного параллелепипеда с длиной 5, шириной 4 и высотой 3. Разработать модель данных, основанную на иерархии классов. Базовый класс описывает общие поля и методы фигур, в классах наследниках добавляются собственные свойства и описывается поведение.

Критерии оценивания

Оценка	Критерии
Отлично	Знает принципы объектно-ориентированного программирования. Разрабатывает соответствующий задаче класс или классы для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Умеет создавать объекты классов и работать с ними. Понимает принципы инкапсуляции и правила определения доступа к элементам класса. Создает полноценное консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости.
Хорошо	Знает принципы объектно-ориентированного программирования. Разрабатывает соответствующий задаче класс или классы для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Умеет создавать объекты классов и работать с ними. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа. Создает полноценное консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости с небольшими неточностями
Удовлетворительно	Знает принципы объектно-ориентированного

	программирования. Разрабатывает соответствующий задаче класс или классы для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Некоторые методы спроектированных недостаточно верно. Умеет создавать объекты классов и работать с ними. Не вполне понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа. Создает полноценное консольное приложение в среде разработки. С трудом поясняет и изменяет код.
Неудовлетворительно	Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса.. Не умеет создавать и работать с объектами классов. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса. Не может создать консольное приложение в среде разработки или создает неработающий проект. Не может пояснить код и изменить его.

Примеры заданий по теме «Обработка текстовой информации с использованием класса String»:

1. Вводится предложение, слова разделены пробелами. Расположить слова в нём в порядке возрастания числа букв в словах.

Возможные тесты

Входные данные	Результат
He likes classical music	He likes music classical
They are on holiday	on are They holiday
Are on	on Are
on	on

2. Вводится текст телеграммы и стоимость одного слова. Слова разделены пробелами, знаки препинания считаются отдельными словами. Определить стоимость телеграммы, если цена одного слова 2 руб..

Возможные тесты

Входные данные	Результат
В случае неявки, акт осмотра будет составлен без Вашего участия.	24
Текст телеграммы	4
телеграмма	2
«пустая строка»	0 или сообщение об отсутствии текста

Критерии оценивания

Оценка	Критерии
Отлично	Знает класс String и применяет методы этого класса, подходящие для решения задачи. Знает и применяет регулярные выражения. Программа выполняется для всех тестов. Создает полноценное консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости
Хорошо	Знает класс String и применяет методы этого класса,

	подходящие для решения задачи в некоторых случаях с подсказкой преподавателя. Знает и применяет регулярные выражения. Программа выполняется для всех тестов, если какой-либо из тестов не выполняется программа дорабатывается. Создает консольное приложение в среде разработки. Поясняет код и изменяет его при необходимости с небольшими неточностями
Удовлетворительно	Знает класс String, но не может без подсказки выбрать методы и полностью самостоятельно составить алгоритм для решения задачи либо решает только самые простые задачи. Знает регулярные выражения., но составляет их с ошибками и не всегда понимает в каких случаях они применяются Программа выполняется для всех тестов, возможно кроме одного-двух крайних случаев. Создает консольное приложение в среде разработки. С трудом поясняет код, не может изменить код при усложнении или существенном дополнении задачи
Неудовлетворительно	Не умеет создавать и использовать объекты класса String. Не может подобрать подходящие для решения методы этого класса. Не может составить регулярное выражение. Программа не выполняется для большинства тестов. Не может создать консольное приложение в среде разработки или создает неработающий проект. Не может пояснить код и изменить его.

Контрольная работа №1

Вариант 1

Сущность "трехмерный вектор" характеризуется координатами (три действительных числа). Создайте класс для хранения и обработки данных о векторе, где есть методы позволяющие определить модуль вектора, вывести информацию о векторе в текстовом виде, вычислить сумму двух векторов (покоординатно);

Вариант 2

Сущность "человек" характеризуется именем и годом рождения (строка и целое число). Создайте класс для хранения и обработки данных о человеке, где есть метод, определяющий возраст человека в заданном году, метод, сравнивающий двух людей на равенство (совпадение имени и года рождения), метод, выводящий информацию о человеке в текстовом виде;

Вариант 3

Сущность "круг" характеризуется координатами центра и радиусом (три действительных числа). Создайте класс для хранения и обработки данных о круге, где есть методы позволяющие определить площадь круга, вывести информацию о круге в текстовом виде, сравнить два круга на равенство (круги равны, если совпадают их центры и радиусы);

Вариант 4

Сущность "абитуриент" характеризуется именем и суммой баллов (строка и целое число). Создайте класс для хранения и обработки данных об абитуриенте, где есть метод, определяющий проходит ли абитуриент конкурс при заданном проходном балле, метод, сравнивающий двух абитуриентов по количеству баллов, метод, выводящий информацию об абитуриенте в текстовом виде;

Критерии оценивания

Оценка	Критерии
Отлично	Разрабатывает соответствующий задаче класс для

	хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, все алгоритмы реализованы верно. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Умеет создавать объекты классов и работать с ними. Понимает принципы инкапсуляции и правила определения доступа к элементам класса.
Хорошо	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, алгоритмы реализованы верно возможно с небольшими неточностями. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Умеет создавать объекты классов и работать с ними. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа.
Удовлетворительно	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Алгоритмы реализации методов содержат ошибки, но не более двух грубых. Некоторые методы спроектированных недостаточно верно. Умеет создавать объекты классов и работать с ними. Не вполне понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа.
Неудовлетворительно	Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса, ошибается в алгоритмах. Не умеет создавать и работать с объектами классов. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса.

Контрольная работа №2

Вариант 1

Сущность "трехмерный вектор" характеризуется координатами (три действительных числа).

а) создайте класс для хранения и обработки данных о векторе, где есть методы позволяющие определить модуль вектора, вывести информацию о векторе в текстовом виде, вычислить сумму двух векторов (покоординатно);

б) в файле с именем data.txt содержится информация о двух векторах (шесть чисел); создайте два соответствующих объекта, выведите на консоль информацию о каждом векторе, его модуль, а также сумму этих двух векторов.

Вариант 2

Сущность "человек" характеризуется именем и годом рождения (строка и целое число).

а) создайте класс для хранения и обработки данных о человеке, где есть метод, определяющий возраст человека в заданном году, метод, сравнивающий двух людей на равенство (совпадение имени и года рождения), метод, выводящий информацию о человеке в текстовом виде;

б) в файле с именем data.txt содержится информация о двух людях (строка число строка число); создайте два соответствующих объекта, выведите на консоль информацию о каждом человеке, его возраст в текущем году, а также информацию о сравнении этих двух людей.

Вариант 3

Сущность "круг" характеризуется координатами центра и радиусом (три действительных числа).

а) создайте класс для хранения и обработки данных о круге, где есть методы позволяющие определить площадь круга, вывести информацию о круге в текстовом виде, сравнить два круга на равенство (круги равны, если совпадают их центры и радиусы);

б) в файле с именем data.txt содержится информация о двух кругах (шесть чисел); создайте два соответствующих объекта, выведите на консоль информацию о каждом круге, его площадь, а также информацию о сравнении этих двух кругов.

Вариант 4

Сущность "абитуриент" характеризуется именем и суммой баллов (строка и целое число).

а) создайте класс для хранения и обработки данных об абитуриенте, где есть метод, определяющий проходит ли абитуриент конкурс при заданном проходном балле, метод, сравнивающий двух абитуриентов по количеству баллов, метод, выводящий информацию об абитуриенте в текстовом виде;

б) в файле с именем data.txt содержится информация о двух абитуриентах (строка число строка число); создайте два соответствующих объекта, выведите на консоль информацию о каждом абитуриенте в порядке убывания количества баллов, а также проходят ли они конкурс при проходном балле 200.

Критерии оценивания

Оценка	Критерии
Отлично	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, все алгоритмы реализованы верно. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Выполняет ввод и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы. Понимает принципы инкапсуляции и правила определения доступа к элементам класса. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов).
Хорошо	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, алгоритмы реализованы верно возможно с небольшими неточностями. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Выполняет ввод

	и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы с небольшими неточностями. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов), возможно не во всех случаях.
Удовлетворительно	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Алгоритмы реализации методов содержат ошибки, но не более двух грубых. Некоторые методы спроектированных недостаточно верно. Выполняет ввод и вывод данных с ошибками. Реализует подходящие интерфейсы, но с ошибками. Не вполне понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов) с ошибками или реализует собственные методы вместо использования стандартных.
Неудовлетворительно	Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса, ошибается в алгоритмах. Не умеет выполнять ввод и вывод данных. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса. Не умеет реализовывать стандартные интерфейсы и использовать стандартные алгоритмы.

Лабораторная работа по теме «Управление доступом. Инкапсуляция.»

Разработать подходящие классы для решения задачи: вовремя ли читатель сдал книгу в библиотеку?

Поля класса являются закрытыми. В классе должны быть: конструктор по умолчанию, конструктор с параметрами, конструктор копирования, методы возвращающие и изменяющие значения полей, методы для решения задачи. Для проверки вовремя ли сдана книга надо написать отдельный метод (подумайте в каком классе он должен располагаться и какие параметры должны быть у этого метода).

В главном классе приложения создайте двух читателей: Петров, взявший книгу Г. Шилдта «Java 8. Полное руководство» до 1 апреля и Васечкин, взявший книгу Л. Толстого «Крейцера соната» до 25 марта и определите кто сдал книгу вовремя, если оба сдали книги 29 марта.

Подсказка.

Класс книга: возможные поля – название, автор, день сдачи книги, номер месяца сдачи книги.

Класс читатель: возможные поля – имя, книга, взятая в библиотеке.

Другой вариант.

Класс книга: возможные поля – название, автор.

Класс читатель: возможные поля – имя, книга, взятая в библиотеке, день сдачи книги, номер месяца сдачи книги.

Критерии оценивания

Оценка	Критерии
Отлично	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, все алгоритмы реализованы верно. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Понимает принципы инкапсуляции и правила определения доступа к элементам класса.
Хорошо	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, алгоритмы реализованы верно возможно с небольшими неточностями. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа.
Удовлетворительно	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Алгоритмы реализации методов содержат ошибки, но не более двух грубых. Некоторые методы спроектированных недостаточно верно. Не вполне понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа.
Неудовлетворительно	Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса, ошибается в алгоритмах. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса.

Лабораторная работа по теме «Интерфейсы»

Разработать класс "Работник предприятия". Работник характеризуется именем и окладом. Зарботная плата вычисляется по количеству отработанных дней, полный оклад начисляется за 22 отработанных дня, если дней меньше или больше, то пропорционально количеству отработанных дней.

Выполнить следующие действия:

- 1) Создать класс, содержащий необходимые поля и методы для их получения и изменения, метод расчета зарплаты.
- 2) Создать в классе конструкторы: по умолчанию, копирования, с параметрами.
- 3) Переопределить методы String toString() для представления информации об объекте в текстовом виде, boolean equals(объект) для сравнения объектов на равенство по имени.
- 4) Реализовать интерфейс для сравнение объектов на больше или меньше по окладу.
- 5) В главном классе приложения создать три объекта, например: "Иванов" и 15000, "Петров" и 12000, "Сидоров" и 20000.
- 6) Рассчитать и вывести на экран зарплату Иванова за 15 дней работы.
- 7) Вывести информацию о том, у кого самая большая зарплата, у кого самая маленькая. Критерии оценивания

Оценка	Критерии
Отлично	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, все алгоритмы реализованы верно. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Выполняет ввод и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы. Понимает принципы инкапсуляции и правила определения доступа к элементам класса. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов).
Хорошо	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, алгоритмы реализованы верно возможно с небольшими неточностями. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Выполняет ввод и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы с небольшими неточностями. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов), возможно не во всех случаях.
Удовлетворительно	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Алгоритмы реализации методов содержат ошибки, но не более двух грубых. Некоторые методы спроектированных недостаточно верно. Выполняет ввод и вывод данных с ошибками. Реализует подходящие интерфейсы, но с ошибками. Не вполне понимает принципы инкапсуляции и правила

	определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов) с ошибками или реализует собственные методы вместо использования стандартных.
Неудовлетворительно	Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса, ошибается в алгоритмах. Не умеет выполнять ввод и вывод данных. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса. Не умеет реализовывать стандартные интерфейсы и использовать стандартные алгоритмы.

Лабораторная работа по теме «Обработка исключительных ситуаций.»

Решаем задачу определения кандидатов на специальную стипендию для школьников. Класс «Школьник», параметры – имя, возраст и успеваемость. Успеваемость школьника определяется годовыми оценками по предметам и участием в предметных олимпиадах разного уровня (школьная, городская, областная). Специальная стипендия назначается школьникам, если у них оценка 5 за предметы: математика, русский язык, история, английский язык, остальные оценки не ниже 4 и есть участие в областной олимпиаде либо первое место в школьной либо призовое в городской. Данные о школьниках вводятся из файла или с клавиатуры.

Проверить корректность вводимых данных: полноту, порядок, значения: имя начинается с большой буквы, оценки – числа от 2 до 5. В случае ошибки генерировать и обрабатывать исключения. Корректность данных проверять в классе, в конструкторе и методах, которые добавляют или меняют значения полей класса.

Критерии оценивания

Оценка	Критерии
Отлично	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все необходимые методы, все алгоритмы реализованы верно. Каждый метод решает только одну задачу. Среди аргументов методов нет лишних. Выполняет ввод и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы. Правильно в нужных местах применяет технологию генерации и обработки исключений. Понимает принципы инкапсуляции и правила определения доступа к элементам класса. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов). Правильно анализирует и обрабатывает ошибочные и исключительные ситуации
Хорошо	Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся только необходимые поля. Присутствуют все

	<p>необходимые методы, алгоритмы реализованы верно возможно с небольшими неточностями. Каждый метод решает только одну задачу, возможно кроме одного-двух спроектированных недостаточно верно. Выполняет ввод и вывод данных, умеет работать с файлами. Реализует подходящие интерфейсы с небольшими неточностями. Синтаксически правильно обрабатывает исключительные ситуации, возможно с неточностями определяет место генерации или обработки одной из исключительных ситуаций. Понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для одного-двух методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов), возможно не во всех случаях. Правильно определяет все ошибочные и исключительные ситуации за исключением одной – двух.</p>
Удовлетворительно	<p>Разрабатывает соответствующий задаче класс для хранения и обработки данных. В классе содержатся необходимые поля, но могут присутствовать лишние. Присутствуют не все необходимые методы или добавлены явно лишние. Алгоритмы реализации методов содержат ошибки, но не более двух грубых. Некоторые методы спроектированных недостаточно верно. Выполняет ввод и вывод данных с ошибками. Реализует подходящие интерфейсы, но с ошибками. В целом понимает механизм генерации и обработки исключений, но не всегда уместно его использует. Не вполне понимает принципы инкапсуляции и правила определения доступа к элементам класса, возможно для ряда полей и методов неверно назначен уровень доступа. Использует способы сравнения объектов, определенные стандартом языка Java (реализация стандартных интерфейсов и использование стандартных алгоритмов) с ошибками или реализует собственные методы вместо использования стандартных. Не может выделить и обработать часть исключений.</p>
Неудовлетворительно	<p>Не знает или плохо знает принципы объектно-ориентированного программирования. Не может разработать соответствующий задаче класс для хранения и обработки данных. Не может создать метод класса, ошибается в алгоритмах. Не умеет выполнять ввод и вывод данных. Не понимает принципы инкапсуляции и правила определения доступа к элементам класса. Не умеет реализовывать стандартные интерфейсы и использовать стандартные алгоритмы. Не может определить, где в программе возникает исключительная ситуация.</p>

Список вопросов к зачету:

1. Основные понятия объектно-ориентированного программирования.
 1. Объект, взаимодействие объектов
 2. Инкапсуляция
 3. Наследование
 4. Полиморфизм
2. Синтаксис языка Java.
 1. Переменные, литералы.
 2. Типы данных.
 3. Операторы.
 4. Описание класса.
 5. Описание полей и методов.
 6. Подключение и использование пакетов.
 7. Массивы.
 8. Методы, передача данных в методы
3. Классы
 1. Поля и методы
 2. Доступ к членам класса.
 3. Объекты, объектные ссылки.
 4. Работа с объектами: создание, вызов методов, сравнение.
4. Наследование.
 1. Правила создания и использования классов-потомков.
 2. Доступ к членам класса-предка.
 3. Наследование методов, переопределение и перегрузка.
 4. Интерфейсы. Реализация интерфейсов
 5. Абстрактные классы
5. Обработка исключительных ситуаций
 1. Генерация исключений
 2. Блок перехвата исключений и блок-ловушка
 3. Стандартные исключения.
 4. Пользовательские исключения.
6. Ввод/вывод.
 1. Поточковые классы для ввода/вывода.
 2. Работа с файлами.
 3. Класс Scanner.
7. Строки.
 1. Обработка текстовой информации с использованием класса String.
 2. Основные методы класса, операции со строками.
8. Классы-обертки, обертки примитивных типов

Методические указания по выставлению зачета

Зачет выставляется по результатам выполнения всех самостоятельных, лабораторных и контрольных работ на оценку не ниже удовлетворительно. Работы выполняются и сдаются в течение семестра последовательно в процессе освоения материала или в исключительных случаях на зачете. В случае необходимости преподаватель в ходе сдачи работ может провести беседу по вопросам к зачету, связанным в первую очередь с тематикой работы.

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования,

описание шкалы оценивания

2.1. Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы

формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объёме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;

- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка «зачтено», «незачтено».

Показатели и критерии, используемые при выставлении оценки подробно описаны в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций».

Высокий уровень формирования компетенций соответствует оценке «отлично» за самостоятельные, лабораторные, контрольные работы.

Оценка «отлично» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована на высоком уровне.

Оценка «хорошо» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на продвинутом уровне.

Оценка «удовлетворительно» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «неудовлетворительно» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Объектно-ориентированное программирование»

Методические указания для студентов по освоению дисциплины

Основной формой изложения учебного материала по дисциплине «Объектно-ориентированное программирование» в первой половине курса являются лекции, так как изучение языка программирования требует знания определенного количества теоретического материала, причем в достаточно большом объеме. Во второй половине курса большинство занятий представляют из себя лекции-беседы, на которых обсуждаются технологии программирования и конкретные примеры их реализации, а так же мастер-классы, где преподаватель демонстрирует процесс разработки программного обеспечения, основные его аспекты и обсуждает возможности рассматриваемых инструментов разработки.

Для успешного освоения дисциплины очень важно решение достаточно большого количества задач, как в аудитории, так и самостоятельно в качестве домашних заданий. В основном такими задачами являются лабораторные работы различного объема, а так же небольшие задачи связанные с исправлением ошибок и доработкой программ. Примеры решения задач разбираются на лекциях, кроме того проводятся лабораторные занятия на сопутствующих практикумах ЭВМ: «Практикум на ЭВМ по объектно-ориентированному программированию», «Практикум на ЭВМ по программированию». Для решения всех задач необходимо знать и понимать лекционный материал. Поэтому в процессе изучения дисциплины рекомендуется регулярное повторение пройденного лекционного материала. Материал, законспектированный на лекциях, необходимо дома еще раз прорабатывать и при необходимости дополнять информацией, полученной на консультациях, практических занятиях или из учебной литературы.

Для усвоения материала необходимо в течение всего курса выполнять задания для самостоятельной работы. Кроме того, материал курса соответствует лабораторным работам практикумов: «Практикум на ЭВМ по объектно-ориентированному программированию», «Практикум на ЭВМ по программированию».

Для проверки и контроля усвоения теоретического материала, приобретенных практических навыков разработки программ, в течение обучения проводятся мероприятия текущей аттестации в виде письменных контрольных работ. Также проводятся консультации (при необходимости) по разбору заданий, которые вызвали затруднения.

В конце изучения дисциплины студенты сдают зачет. Зачет выставляется по итогам работы, которая включает в себя выполнение лабораторных и контрольных работ.

Освоить вопросы, излагаемые в процессе изучения дисциплины «Объектно-ориентированное программирование» самостоятельно студенту довольно сложно. Это связано со сложностью изучаемого материала и большим объемом курса. Поэтому посещение всех аудиторных занятий является совершенно необходимым. Без упорных и регулярных занятий в течение семестра сдать зачет по итогам изучения дисциплины студенту практически невозможно.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

Для самостоятельной работы особенно рекомендуется использовать учебную литературу, указанную в разделе № 7 данной рабочей программы.

Также для подбора учебной литературы рекомендуется использовать широкий спектр интернет-ресурсов:

1. Электронно-библиотечная система «Университетская библиотека online» (www.biblioclub.ru) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (*регистрация в электронной библиотеке – только в сети

университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).

2. Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:

1. Личный кабинет (http://lib.uniyar.ac.ru/opac/bk_login.php) дает возможность получения on-line доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.

2. Электронная библиотека учебных материалов ЯрГУ (http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/паролю.

3. Электронная картотека [«Книгообеспеченность»](#) (http://www.lib.uniyar.ac.ru/opac/bk_bookreq_find.php) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека [«Книгообеспеченность»](#) доступна в сети университета и через Личный кабинет.