МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет компьютерных технологий и прикладной математики



«31» мая 2019 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ) Б1.О.31 ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Направление подготовки/специальность *01.03.02 Прикладная математика и информатика*

Направленность (профиль) / специализация *Математическое моделирование в естествознании и технологиях*

Форма обучения *очная*

Квалификация *бакалавр*

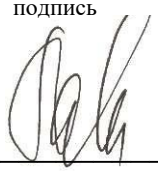
Краснодар 2019

Рабочая программа дисциплины «Объектно-ориентированное программирование» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки / специальности 01.03.02 Прикладная математика и информатика

Программу составил(и):



В.В. Подколзин, доцент, канд. физ.-мат. наук _____
И.О. Фамилия, должность, ученая степень, ученое звание подпись



О.В. Гаркуша, доцент, канд. физ.-мат. наук, доцент _____
И.О. Фамилия, должность, ученая степень, ученое звание подпись

Рабочая программа дисциплины «Объектно-ориентированное программирование» утверждена на заседании кафедры информационных технологий протокол № 15 от «07» мая 2019 г.



И. о. зав. кафедрой (разработчика) О.В. Гаркуша _____
фамилия, инициалы подпись

Рабочая программа обсуждена на заседании кафедры утверждена на заседании кафедры математического моделирования протокол № 11 от «10» апреля 2019 г.



Заведующий кафедрой (выпускающей) Бабешко В.А. _____
фамилия, инициалы подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 1 от «15» мая 2019г.



Председатель УМК факультета Коваленко А.В _____
фамилия, инициалы подпись

Рецензенты:

Рубцов Сергей Евгеньевич, кандидат физико-математических наук, доцент кафедры математического моделирования ФГБГОУ «КубГУ»

Бегларян Маргарита Евгеньевна, кандидат физико-математических наук, доцент, заведующий кафедрой СГЕНД СКФ ФГБОУ ВО «Российский государственный университет правосудия»

1 Цели и задачи изучения дисциплины.

1.1 Цель освоения дисциплины.

Цель дисциплины «Программирование на Java» — изучение студентами объектно-ориентированного подхода в программировании. Все практические примеры рассматриваются на языке программирования Java.

Воспитательная цель: формирование свободного и творческого подхода к программированию на современных языках высокого уровня, интереса к наблюдению за тенденциями и новостями в области средств разработки программного обеспечения.

Основные задачи курса на основе системного подхода:

- введение в теорию разработки кросс-платформенных приложений;
- знакомство с компилятором JDK и средой разработки NetBeans и сравнение ее с другими платформами разработки ПО для ОС Microsoft Windows;
- изучение Java как языкового средства, наиболее полно отражающего возможности создания кросс-платформенных приложений;
- формирование навыков создания приложений на языке Java;
- совершенствование и углубление навыков программирования, изучение последних нововведений в области ООП, реализованных в Java;

Отбор материала основывается на необходимости ознакомить студентов со следующей современной научной информацией:

- об особенностях и последних достижениях в области разработки кросс-платформенного ПО;
- о положительных и отрицательных чертах подхода к программированию, реализованному в языке Java.

Содержательное наполнение дисциплины обусловлено общими задачами в подготовке бакалавра.

Научной основой для построения программы данной дисциплины является теоретико-прагматический подход в обучении.

1.2 Задачи дисциплины.

- закрепление навыков алгоритмизации и программирования на основе углубленного изучения объектно-ориентированного подхода в программировании;
- знакомство с принципами инкапсуляции, наследования и полиморфизма.
- обучение созданию мультиплатформенных приложений.

1.3 Место дисциплины в структуре образовательной программы.

Дисциплина «Программирование на Java» относится к вариативной части Блока 1 "Дисциплины (модули)" учебного плана.

Дисциплина «Программирование на Java» является логически и содержательно-методически связана с такими дисциплинами как «Язык программирования C++», «Программирование на основе API», «Языки программирования и методы трансляции». Является логически связанной с математическими дисциплинами, рассматривает объекты таких дисциплин как «Математическая логика и дискретная математика», «Теория графов и ее приложения» с точки зрения программирования.

Входными знаниями для освоения данной дисциплины являются знания, умения и опыт, накопленный студентами в процессе изучения дисциплины «Язык программирования C++», «Языки программирования и методы трансляции».

1.4 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.

Изучение данной учебной дисциплины направлено на формирование у обучающихся профессиональных компетенций (ПК)

1. Знать основные алгоритмические и программные решений разработки кросс-платформенных приложений;
2. Знать принципы разработки системного и программного обеспечения с помощью Java;
3. Уметь в рамках профессиональной деятельности применять язык программирования Java для разработки решений в области системного и программного обеспечения;
4. Уметь объяснять принципы алгоритмических решений своих программ;
5. Уметь участвовать в проектировании и разработке программного обеспечения;
6. Владеть навыками алгоритмических и программных решений современного объектно-ориентированного программирования;
7. Владеть способность к разработке и применению кросс-платформенных приложений;

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			Знать	**Уметь**	**Владеть**
1.	ПК-7	способностью к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения	1, 2	3, 4, 5	6, 7

2. Структура и содержание дисциплины.

2.1 Распределение трудоёмкости дисциплины по видам работ.

Общая трудоёмкость дисциплины составляет 5 зач. ед. (180 часов), их распределение по видам работ представлено в таблице

Вид учебной работы		Всего часов	Семестры (часы)			
			7	___		
Контактная работа, в том числе:						
Аудиторные занятия (всего):						
Занятия лекционного типа		36	36	-	-	-
Лабораторные занятия		36	36	-	-	-
Занятия семинарского типа (семинары, практические занятия)		-	-	-	-	-
		-	-	-	-	-
Иная контактная работа:						
Контроль самостоятельной работы (КСР)		8	8			
Промежуточная аттестация (ИКР)		0,5	0,5			
Самостоятельная работа, в том числе:						
Курсовая работа		-	-	-	-	-
Проработка учебного (теоретического) материала		20	20	-	-	-
Выполнение индивидуальных заданий		30	30	-	-	-
Реферат		-	-	-	-	-
Подготовка к текущему контролю		4,8	4,8	-	-	-
Контроль:						
Подготовка к экзамену		44,7	44,7			
Общая трудоемкость	**час.**	**180**	**180**	**-**	**-**	**-**
	в том числе контактная работа	**80,5**	**80,5**			
	зач. Ед	**5**	**5**			

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы дисциплины, изучаемые в 7 семестре *(очная форма)*

№	Наименование разделов		Количество часов			
		Всего	Аудиторная Работа		Внеаудиторная работа	
			Л	ЛР	СРС	контроль
м1	2	3	4	5	6	7
1.	Введение в язык Java, среда программирования NetBeans	6	2		2	2
2.	Основные конструкции языка Java	12	2	2	4	4
3.	Объекты и классы	22	6	6	6	4
4.	Наследование	18	4	4	6	4
5.	Интерфейсы	18	4	4	6	4
6.	Программирование графики	19	4	4	7	4
7.	Обработка событий	18	4	4	6	4
8.	Исключения	16	4	4	4	4
9.	Потоки	28	6	6	8	8
10.	Подготовка к сдаче и сдача зачета	14,5		2	5,8	6,7
	Контроль самостоятельной работы (КСР)	8				

	Промежуточная аттестация (ИКР)	0,5				
	ИТОГО	180	36	36	54,8	44,7

Примечание: Л – лекции, ПЗ – практические занятия / семинары,ЛР – лабораторные занятия, КСР – контролируемая самостоятельная работа, СРС – самостоятельная работа студента

2.3 Содержание разделов дисциплины:

2.3.1 Занятия лекционного типа.

№	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
1.	Введение в язык Java, среда программирования NetBeans	Программная платформа Java, характерные особенности языка Java, история языка, выбор среды разработки программ, использование инструментов, вызываемых из командной строки, использование интегрированной среды разработки программ NetBeans.	Индивидуальные задания
2.	Основные конструкции языка Java	Типы данных, переменные, константы, операторы, строки, массивы, блоки, условные выражения, циклы.	Индивидуальные задания
3.	Объекты и классы	Использование встроенных классов, определение собственных классов, поля и методы классов, инкапсуляция, конструкторы, перегрузка методов, пакеты.	Индивидуальные задания
4.	Наследование	Классы, суперклассы, подклассы, иерархия, полиморфизм, динамическое связывание, глобальный суперкласс Object.	Индивидуальные задания
5.	Интерфейсы	Интерфейсы, свойства интерфейсов, клонирование объектов, обратный вызов.	Индивидуальные задания
6.	Программирование графики	Пакет Swing, создание и позиционирование фреймов, отображение информации на панели, двумерные фигуры, работа с цветом, вывод текста.	Индивидуальные задания
7.	Обработка событий	Общие сведения об обработке событий, классы-адаптеры, иерархия событий библиотеки AWT, семантические и низкоуровневые события, действия.	Индивидуальные задания
8.	Исключения	Классификация исключений, объявление контролируемых исключений, генерация исключений, создание классов исключений, перехват исключений, регистрация ошибок.	Индивидуальные задания
9.	Потоки	Потоки выполнения, синхронизация потоков, планирование потоков.	Индивидуальные задания

2.3.2 Занятия семинарского типа – не предусмотрены.

2.3.3 Лабораторные занятия.

№	Наименование лабораторных работ	Форма текущего контроля
1	3	4
1.	Практическая реализация работы с примитивными типами данных в Java. Работа с классом String и классом Scanner.	Индивидуальные задания
2.	Решение задач на языке программирования Javac использованием классов. Создание конструкторов класса. Применение техники совмещения методов. Работа со встроенными пакетами. Создание собственных пакетов.	Индивидуальные задания
3.	Решение задач с использованием механизма наследования. Применение техники замещения методов. Работа с абстрактными классами	Индивидуальные задания
4.	Решение задач с использованием механизма интерфейсов. Сравнение интерфейсов и абстрактных классов.	Индивидуальные задания
5.	Использование пакетов Swing и AWTдля создания оконных приложений. Работа с фреймами, панелями. Программирование графики с использованием класса Graphics2D	Индивидуальные задания
6.	Решение задач, использующих механизм обработки событий клавиатуры и мыши	Индивидуальные задания
7.	Доработка задач на механизм наследования и механизм интерфейсов с использованием встроенных и собственных классов исключительных ситуаций	Индивидуальные задания
8.	Разработка многопоточных приложений, управление работой потока, изучение механизмов синхронизации потоков, изучение ситуаций взаимной блокировки потоков	Индивидуальные задания

2.3.4 Курсовые работы – не предусмотрены

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Выполнение индивидуальных заданий	Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие. Ю.В. Кольцов [и др.] – Краснодар: Кубанский гос.ун-т, 2017 Синица С.Г. Уварова А.В. Программирование на Java: учебное пособие. КубГУ, Краснодар, 2016 г. ISBN: 978-5-8209-1215-3
2	Проработка учебного (теоретического) материала	Синица С.Г. Уварова А.В. Программирование на Java: учебное пособие. КубГУ, Краснодар, 2016 г. ISBN: 978-5-8209-1215-3 Николаев, Е.И. Объектно-ориентированное программирование : учебное пособие/ Е.И. Николаев ; Министерство образования и науки Российской Федерации,

		Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет». - Ставрополь : СКФУ, 2015. - 225 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=458133

3. Образовательные технологии.

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

– Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

– Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

– Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

– Информационно-коммуникационные технологии (ИКТ) - расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

– Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные

процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
7	Л, ЛР	Практические занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	26
Итого			26

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации.

4.1Фонд оценочных средств для проведения текущего контроля.

В качестве оценочных средств текущего контроля успеваемости и промежуточной аттестации используются: коллоквиум, индивидуальные практические задания.

Перечень задач текущего контроля по темам:

Перечень компетенций, проверяемых оценочным средством:

ПК-7 -способностью к разработке и применению алгоритмических и программных решений в области системного и программного обеспечения

Задачи по теме «Объекты и классы»,

1. Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и вращения на заданный угол. Написать программу, демонстрирующую работу с этими классами.
2. Построить описание класса, содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса. Написать программу, демонстрирующую работу с этим классом.
3. Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания, умножения комплексных чисел. Написать программу, демонстрирующую работу с этим классом.
4. Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания с получением нового вектора, вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами. Написать программу, демонстрирующую работу с этим классом.
5. Составить описание класса для определения одномерных массивов целых чисел (векторов). Предусмотреть возможность обращения к отдельному элементу массива с контролем выхода за пределы массива, возможность задания произвольных границ индексов при создании объектов, возможность поэлементного сложения и вычитания массивов с одинаковыми границами, умножения и деления всех элементов массива на скаляр, вывода на экран всего массива. Написать программу, демонстрирующую работу с этим классом.
6. Составить описание класса для определения одномерного массива строк фиксированной длины. Предусмотреть возможность обращения к отдельным строкам массива по индексам, контроль выхода за границы массива, поэлементное сцепление двух массивов с образованием нового массива с исключением повторяющихся элементов, вывод на экран элемента по индексу и всего массива. Написать программу, демонстрирующую работу с этим классом.
7. Составить описание класса многочленов от одной переменной, задаваемых степенью многочлена и массивом коэффициентов. Предусмотреть методы для вычисления значения многочлена для заданного аргумента, операции сложения, вычитания, умножения многочленов с получением нового объекта-многочлена, вывод на экран описания многочлена. Написать программу, демонстрирующую работу с этим классом.
8. Составить описание класса, обеспечивающего представление матрицы произвольного размера с возможностью изменения числа строк и столбцов, вывода

на экран подматрицы любого размера и всей матрицы. Написать программу, демонстрирующую работу с этим классом.

9. Описать класс «домашняя библиотека». Предусмотреть возможность работы с произвольным числом книг, поиска книги по какому-либо признаку, добавления книг в библиотеку, удаления из нее, сортировки книг по разным полям. Написать программу, демонстрирующую работу с этим классом.
10. Описать класс «записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку, добавления и удаления записей, сортировки по разным полям. Написать программу, демонстрирующую работу с этим классом.

Задачи по теме «Наследование»

1. Определить иерархию классов: точка на плоскости – окружность – эллипс. Методы – инициализация, вывод на экран, вычисление периметра и площади.
2. Определить иерархию классов: четырехугольник – прямоугольник – квадрат. Методы – инициализация, вывод на экран, вычисление периметра и площади.
3. Класс1: корабль; поля – название, водоизмещение, длина, координаты на плоскости, крейсерская скорость. Методы – назначение полей, печать, перемещение корабля на плоскости. Класс2: пассажирский корабль; поля – количество мест, количество шлюпок. Методы – расчет количества спасшихся пассажиров при крушении. Класс3: военный корабль; поля – количество орудий; максимальная дальность поражения мишени; методы – расчет возможности поражения цели в заданной точке.
4. Определить класс «Студент», экземпляры которого имеют поля фамилия, имя, курс и идентификационный номер. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – «Студент-дипломник», экземпляры которого имеют тему диплома. Определить функцию вывода на экран. Определить функцию переназначения темы диплома.
5. Определить класс «Автомобиль», экземпляры которого имеют поля марка, мощность, объем двигателя. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – такси, экземпляры которого имеют поле – тариф. Определить функцию вывода на экран. Определить функцию переназначения тарифа и функцию расчета стоимости поездки.
6. Определить класс «Жидкость», экземпляры которого имеют поля название, плотность. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – раствор кислоты, экземпляры которого имеют поле – концентрация. Определить функцию вывода на экран. Определить функцию переназначения концентрации.
7. Определить класс «Автомобиль», экземпляры которого имеют поля – марка, мощность, объем двигателя. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – грузовой автомобиль, экземпляры которого имеют поле – грузоподъемность. Определить функцию вывода на экран. Определить функцию переназначения грузоподъемности и функцию, определяющую возможность перевозки груза.
8. Определить класс «Работник», и производные классы – «Служащий с почасовой оплатой», «Служащий в штате», «Служащий с процентной ставкой». Определить функции назначения и изменения полей и функцию вывода на экран и функцию назначения заработной платы.

9. Определить класс «Корабль», экземпляры которого имеют поля название, водоизмещение, координаты на плоскости. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – «Военный корабль», экземпляры которого имеют поле – количество орудий. Определить функцию вывода на экран. Определить 2 функции перемещения военного корабля на плоскости.
10. Определить класс «Самолет», экземпляры которого имеют поля – марка, объем топливного бака, максимальная дальность полета. Определить функции назначения и изменения полей и функцию вывода на экран. Определить производный класс – «Пассажирский самолет», экземпляры которого имеют поле – количество посадочных мест. Определить функцию вывода на экран. Определить функцию переназначения количества мест.

Задачи по теме «Интерфейсы»

1. Определить интерфейс Figure с методами вычисления периметра и площади. Реализовать интерфейс в классах Rectangle, Circle, Trapezium.
2. Определить интерфейс Number с методами – арифметическими операциями. Реализовать интерфейс в классах Integer и Real.
3. Определить интерфейс Body с методами вычисления площади поверхности и объема. Реализовать интерфейс в классах Parallelepiped и Ball.
4. Определить интерфейс Triangle с методами вычисления периметра и площади. Треугольник должен задаваться 2 сторонами и углом между ними. Реализовать интерфейс в классах прямоугольный треугольник, равнобедренный треугольник, равносторонний треугольник.
5. Определить интерфейс Root с методами вычисления вычисления корней и вывода результата на экран. Реализовать интерфейс в классах Linear и Square.
6. Определить интерфейс Function с методами вычисления функции в заданной точке и вывода результата на экран. Реализовать интерфейс в классах Ellipse и Hyperbola.
7. Определить интерфейс Pair с методами арифметическими операциями. Реализовать интерфейс в классах Complex и Rational (рациональная дробь).
8. Определить интерфейс Triad с методами увеличения на 1. Реализовать интерфейс в классах Date и Time.
9. Определить интерфейс Pair с методами арифметическими операциями. Реализовать интерфейс в классах Money и Fraction (дробное число).
10. Определить интерфейс Integer с методами арифметическими операциями и выводом на экран. Реализовать интерфейс в классах Decimal и Binary. Число представляется массивом, каждый элемент которого цифра.

Задачи по теме «Программирование графики»

1. Прямоугольная область задана координатами x1, y1, x2 y2 концов ее диагонали. Область разбита на прямоугольники так, что одна сторона разбита на n, а другая на m отрезков. В этой области задан треугольник вершинами u1, v1, u2, v2, u3, v3. Вычислить количество прямоугольников области, в которых лежит хотя бы одна точка треугольника. Выделить искомые прямоугольники.
2. Прямоугольник задается координатами левого верхнего угла, шириной и высотой. Окружность задается координатами центра и радиусом. Треугольник задается координатами вершин. Выделить область пересечения этих фигур.

3. Три точки задаются координатами x1, y1, x2 y2, x3, y3. Определить, возможно ли построить по этим точкам треугольник, и если это возможно, вписать в этот треугольник окружность.
4. Окружность задается координатами центра и радиусом, треугольник задается координатами своих вершин. Также задан произвольный пятиугольник. Выделить область пересечения этих фигур.
5. Нарисовать три окружности, заданные координатами центра и радиусом(A, B, C). Выделить цветом: $A \cap B \cap C$; $A \cap B \backslash C$; $A \backslash B \cup C$.
6. Нарисовать треугольник, заданный координатами своих вершин. Если треугольник прямоугольный, то закрасить его штриховкой. Если треугольник тупоугольный – нарисовать биссектрису тупого угла. Если треугольник остроугольный – подсчитать его площадь.
7. Прямоугольная область задана координатами x1, y1, x2 y2 концов ее диагонали. Область разбита на прямоугольники так, что одна сторона разбита на n, а другая на m отрезков. В этой области задан эллипс. Вычислить количество прямоугольников области, в которых лежит хотя бы одна точка эллипса. Выделить искомые прямоугольники.
8. Нарисовать три треугольника, заданные координатами своих вершин. Выделить область пересечения этих треугольников.
9. Прямоугольная область задана координатами x1, y1, x2 y2 концов ее диагонали. Область разбита на прямоугольники так, что одна сторона разбита на n, а другая на m отрезков. В этой области задан произвольный пятиугольник. Вычислить количество прямоугольников области, в которых лежит хотя бы одна точка пятиугольника. Выделить искомые прямоугольники.
10. Нарисовать ромб, заданный стороной и двумя углами, треугольник, заданный координатами вершин, и эллипс. Выделить область пересечения этих фигур.

Задачи по теме «Обработка событий»

Решить следующую задачу, реализовав пользовательский интерфейс GUI и различные обработчики событий.

1. Описать класс, реализующий тип данных «вещественная матрица» и работу с ним. Класс должен реализовывать следующие операции: сложение, вычитание, умножение, вычисление обратной и транспонированной матрицы, методы вычисления детерминанта и нормы, методы, реализующие проверку типа матрицы. Написать программу, демонстрирующую работу с этим классом.
2. Описать класс множество, позволяющий выполнять основные операции – добавление и удаление элемента, объединение, пересечение и разность множеств. Написать программу, демонстрирующую работу с этим классом.
3. Описать класс, реализующий бинарное дерево, обладающий возможностью добавления новых элементов, удаления существующих, поиска элемента по ключу, а также последовательного доступа ко всем элементам. Написать программу, использующую этот класс для англо-русского словаря.
4. Описать класс для эффективной работы со строками, позволяющий форматировать и сравнивать строки, хранить в строках числовые значения и извлекать их. Написать программу, демонстрирующую работу с этим классом.
5. Описать класс, реализующий стек. Написать программу, использующую этот стек для отыскания прохода по лабиринту. Лабиринт представляется в виде матрицы, состоящей из квадратов. Каждый квадрат либо открыт, либо закрыт. Вход в закрытый квадрат запрещен. Если квадрат открыт, то вход в него возможен со стороны, а не с угла. Каждый квадрат определяется его координатами в матрице.

После отыскания прохода программа печатает найденный путь в виде координат квадратов.

6. Описать класс параллелепипед. Предусмотреть методы создания объектов, перемещения, изменения размеров, вычисления высоты, нахождения центра тяжести. Написать программу, демонстрирующую работу с этим классом.
7. Описать класс «телефонный справочник». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку, добавления и удаления записей, сортировки по разным полям. Написать программу, демонстрирующую работу с этим классом.
8. Описать класс «список». Предусмотреть возможность добавления и удаления элементов в начало, конец списка, в указанную позицию, сортировку по убыванию и возрастанию, подсчет количества элементов в списке, вывод на экран части списка с указанием границ вывода.

Задачи по теме «Потоки»

1. Вычислить сумму элементов массива, используя блочный подход.
2. Вычислить сумму элементов массива, используя циклический подход.
3. Вычислить минимальный элемент матрицы, используя блочный подход.
4. Вычислить максимальный элемент матрицы, используя циклический подход.
5. Решить СЛАУ методом Гаусса.
6. Реализовать LU разложение матриц.
7. Реализовать умножение матриц.
8. На основе программы для решения СЛАУ методом Гаусса напишите параллельную программу для вычисления определителя и постройте зависимость коэффициента ускорения от числа ядер (потоков).
9. Реализовать транспонирование матрицы $n \times n$.
10. Реализовать метод Гаусса (метод элементарных преобразований) для вычисления ранга произвольной матрицы A – $m \times n$.

Перечень вопросов текущего контроля коллоквиума

Перечень компетенций, проверяемых оценочным средством:

ПК-7 -способностью к разработке и применению алгоритмических и программных решений в области системного и программного обеспечения

1. Основные особенности языка программирования Java.
2. Перечислить основные принципы ООП. (Сколько их? Какие? Коротко описать каждый принцип)
3. Что такое кроссплатформенность? Каким образом она достигается в Java?
4. Описать по действиям процесс создания программы на Java с помощью среды разработки NetBeans.
5. Простейшие операторы языка Java.
6. Перечислить простейшие типы в Java.
7. Что такое выражение (expression), утверждение (statement) и блок (block) в Java?
8. Описать понятие класса. (Что это такое? Какова структура класса? Что такое экземпляр класса, как его можно получить?)
9. Чем отличаются друг от друга Статические атрибуты класса, Атрибуты класса и Локальные переменные.
10. Что такое Конструктор? Как объявляется Конструктор в Java классе?

11. Что такое анонимный класс и анонимный объект. Примеры.
12. Что такое сборщик мусора?
13. Что такое "Блок статической инициализации класса"?
14. Что такое вложенный класс? Как из класса B вложенного в класс A получить доступ к атрибутам класса A?
15. Какие способы задания наследования в Java вы знаете? Примеры.
16. Что такое аннотации в Java?
17. Что такое интерфейс?
18. Чем отличается Абстрактный класс от Интерфейса?
19. Общая схема обработки исключений в Java.
20. Генерация собственных исключений, операторы throw и throws

4.2 Фонд оценочных средств для проведения промежуточной аттестации.

Форма промежуточной аттестации – зачет, экзамен.

Для получения зачета студенту необходимо выполнить в течение семестра 5 индивидуальных задачи по темам «Классы», «Наследование», «Интерфейсы», «Программирование графики», «Потоки». Зачет выставляется студентам, сдавшим в течение семестра 5 индивидуальных задач в установленные сроки.

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет, экзамен)

Перечень компетенций, проверяемых оценочным средством:

ПК-7 -способностью к разработке и применению алгоритмических и программных решений в области системного и программного обеспечения

Перечень вопросов для подготовки к экзамену

1. Введение в Java. Характерные особенности языка Java. Запуск программ. Функция main.
2. Типы данных.
3. Переменные. Константы.
4. Основные операции.
5. Математические функции и константы.
6. Строки.
7. Ввод и вывод.
8. Блоки и основные операторы.
9. Одномерные и двумерные массивы.
10. Объекты и классы. Основные понятия.
11. Перегрузка методов.
12. Наследование.
13. Переопределение методов.
14. Динамическое назначение методов.
15. Абстрактные классы.
16. Пакеты.
17. Интерфейсы.
18. Клонирование.
19. Программирование оконных приложений.

20. Программирование графики.
21. Двумерные примитивы библиотеки Java.geom.
22. Общая схема обработки исключений.
23. Обработка исключений операторы try, catch, finally.
24. Обработка исключений операторы throw, throws.
25. Общая схема обработки событий.
26. Использование внутренних классов для обработки событий.
27. Основные классы и интерфейсы для обработки событий.
28. Классы-адаптеры.
29. Потоки. Основные понятия.
30. Синхронизация потоков.

Задачи для подготовки к экзамену

1. Создать класс точка, предусмотреть в нем несколько конструкторов, методы определения координатной четверти, которой принадлежит точка, перемещения точки, метод проверяющий являются ли две точки симметричными относительно начала координат, метод, вычисляющий расстояние между двумя точками.

2. Класс Равнобедренная трапеция, члены класса: координаты 4-х точек. Предусмотреть в классе конструктор и методы: проверка, является ли фигура равнобедренной трапецией; вычисления и вывода сведений о фигуре: длины сторон, периметр, площадь. В функции main продемонстрировать работу с классом: дано N трапеций, найти количество трапеций, у которых площадь больше средней площади.

3. Создать класс Money для работы с денежными суммами. Число должно быть представлено 2 полями: для рублей и для копеек. Копейки при выводе на экран должны отделяться от рублей запятой. Реализовать сложение, вычитание, деление сумм, деление суммы на дробное число, умножение на дробное число и операции сравнения.

4. Создать класс Angle для работы с углами на плоскости, задаваемые в градусах и минутах. Обязательно должны быть реализованы перевод в радианы, приведение к диапазону 0-360, увеличение и уменьшение угла на заданную величину, получение синуса, сравнение углов.

5. Создать класс Date для работы с датами в формате «год.месяц.день». Дата представляется структурой с тремя полями. Класс должен включать не менее трех методов инициализации: числами, строкой и датой. Обязательными операциями являются: вычисление даты через заданное количество дней, вычитание заданного количества дней из даты, определение високосности года, присвоение и получение отдельных частей, сравнение дат, вычисление количества дней между датами.

6. Создать класс Time для работы с временем в формате «час.минута.секунда». Дата представляется структурой с тремя полями. Класс должен включать не менее трех методов инициализации: числами, строкой и временем. Обязательными операциями являются: вычисление разности между двумя моментами времени в

секундах, сложение времени и заданного количества секунд, вычитание заданного количества секунд из времени, сравнение моментов времени, перевод в секунды, перевод в минуты.

7. Реализовать класс Account, представляющий собой банковский счет. В классе должны быть 4 поля: фамилия владельца, номер счета, процент начисления и сумма в рублях. Открытие нового счета выполняется операцией инициализации. Необходимо выполнять следующие операции: сменить владельца, снять некоторую сумму денег, положить деньги, начислить проценты, перевести сумму в доллары,

Критерием оценивания результатов освоения дисциплины (зачет) являются результаты текущего контроля. В случае неудовлетворительных результатов по коллоквиуму или какой-либо индивидуальной задаче, студенту предоставляется возможность повторной сдачи соответствующего элемента контроля.

Критерии оценки:

- **оценка «зачтено»**: студент получил не менее 4 баллов за коллоквиум и не менее 1 балла за каждую из 6 индивидуальных задач.

- **оценка «незачет»**: студент получил менее 4 баллов за коллоквиум или менее 1 балла хотя бы за одну из 6 индивидуальных задач.

Оценка	
Незачет	**Зачтено**
• студент получил менее 4 баллов за коллоквиум • менее 1 балла хотя бы за одну из 6 индивидуальных задач	• студент получил не менее 4 баллов за коллоквиум; • не менее 1 балла за каждую из 6 индивидуальных задач;

Методические рекомендации определяющие процедуры оценивания на экзамене

Итоговой формой контроля сформированности компетенций у обучающихся по дисциплине является экзамен. Студенты обязаны сдать экзамен в соответствии с расписанием и учебным планом

ФОС промежуточной аттестации состоит из вопросов и задач к экзамену по дисциплине.

Экзамен по дисциплине преследует цель оценить работу студента за курс, получение теоретических знаний, их прочность, развитие творческого мышления, приобретение навыков самостоятельной работы, умение применять полученные знания для решения практических задач.

Форма проведения экзамена: письменно.

Экзаменатору предоставляется право задавать студентам дополнительные вопросы по всей учебной программе дисциплины.

Результат сдачи экзамена заноситься преподавателем в экзаменационную ведомость и зачетную книжку.

Методические рекомендации определяющие процедуры оценивания задачи на экзамене

Компонентом промежуточного контроля по дисциплине «Программирование на Java» является задача в виде программного кода. Максимальное количество баллов, которые студент может получить за правильное решение задачи, составляет 2 балла.

Описание	Баллы
Предоставлен работоспособный программный код, студент может пояснить ход решения, знает назначение команд, может изменить некоторые условия по просьбе преподавателя.	2
Программный код может быть не работоспособен, однако алгоритм решения задачи корректный, студент может пояснить ход решения, знает назначение некоторых команд	1
Программный код не работает, алгоритм решения не верный, студент не знает назначения отдельных команд	0

Критерии оценки:

Оценка			
Неудовлетворительно	**Удовлетворительно**	**Хорошо**	**Отлично**
• Не выполнена задача и не ответил ни на один вопрос билета • непонимание сущности излагаемых вопросов, грубые ошибки в ответе, неуверенные и неточные ответы на дополнительные вопросы экзаменаторов.	• Задача выполнена на 1 балл • частично ответил на два вопроса билета • отвечает на дополнительные вопросы кратко, допуская неточности	• достаточно полно ответил на два вопроса билета • задача выполнена на 1-2 балла • твёрдые и достаточно полные ответы на дополнительные вопросы	• исчерпывающий ответ на 2 вопроса билета, с примерами и пояснениями • задача решена на 2 балла • исчерпывающие, логически последовательные, полные, правильные и конкретные ответы на дополнительные вопросы;

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

5.1 Основная литература:

1. Синица С.Г. Уварова А.В. Программирование на Java: учебное пособие. КубГУ, Краснодар, 2016 г. ISBN: 978-5-8209-1215-3

2. Николаев, Е.И. Объектно-ориентированное программирование : учебное пособие/ Е.И. Николаев ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет». - Ставрополь : СКФУ, 2015. - 225 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=458133

5.2. Дополнительная литература:

1. Хабибулин И.Ш. Самоучитель Java 2: СПб. : БХВ-Петербург, 2007. - 719 с. - ISBN 5941575734

2. Уварова А.В. Основы программирования на Java: тексты лекций. КубГУ, Краснодар, 2010 г.

3. Гаврилов, А.В. Программирование на Java. Конспект лекций. [Электронный ресурс] : учеб. пособие / А.В. Гаврилов, С.В. Клименков, Е.А. Цопа. — Электрон. дан. — СПб. : НИУ ИТМО, 2010. — 130 с. — Режим доступа: http://e.lanbook.com/book/43547

5.3. Периодические издания:

1. Прикладная информатика
2. Проблемы передачи информации
3. Программные продукты и системы
4. Программирование
5. COMPUTATIONAL NANOTECHNOLOGY (ВЫЧИСЛИТЕЛЬНЫЕ НАНОТЕХНОЛОГИИ)
6. COMPUTERWORLD РОССИЯ
7. WINDOWS IT PRO / RE

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.

1. Основы Java для начинающих — URL: https://javabegin.ru/free-oop/

2. Javaдля начинающих | Изучаем Javaпрограммирование – URL: http://study-java.ru/
3. JavaRushонлайн-курс обучения программированию на Java – URL: https://javarush.ru/

7. Методические указания для обучающихся по освоению дисциплины.

По курсу предусмотрено проведение лекционных занятий, на которых дается основной систематизированный материал. После прослушивания лекции рекомендуется самостоятельно выполнить на компьютере программные примеры, приводимые в лекции.

Важнейшим этапом курса является самостоятельная работа студентов. При самостоятельной работе необходимо изучить литературу, приведенную в перечнях выше, для осмысления вводимых понятий, анализа предложенных подходов и методов разработки программ.

Разрабатывая решение новой задачи, студент должен уметь выбрать эффективные и надежные структуры данных для представления информации, подобрать соответствующие алгоритмы для их обработки, учесть специфику языка программирования, на котором будет выполнена реализация. Студент должен уметь выполнять тестирование и отладку алгоритмов решения задач с целью обнаружения, и устранения в них ошибок.

На лабораторных занятиях в качестве систем программирования для решения задач и изучения методов и алгоритмов, приведенных в лекциях, рекомендуется использовать компилятор JDK, а также среды разработки NetBeans или Eclipse. Для эффективного программирования рекомендуется использовать встроенные отладчики.

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.

8.1 Перечень информационных технологий.

– Использование электронных презентаций при проведении лекционных занятий.

8.2 Перечень необходимого программного обеспечения.

– Виртуальная Java – машина (Java Development Kit).
– Интегрированная среда разработки программ (NetBeans, Eclipse).

8.3Перечень информационных справочных систем:

1. Электронная библиотечная система eLIBRARY.RU (http://www.elibrary.ru)/

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

№	Вид работ	Материально-техническое обеспечение дисциплины и оснащенность
1.	Лекционные занятия	Лекционная аудитория, оснащенная презентационной

		техникой (проектор, экран, компьютер/ноутбук) и соответствующим программным обеспечением (ПО) MS Power Point.
2.	Лабораторные занятия	Аудитория, оснащенная персональными компьютерами с установленным программным обеспечением (JDK, NetBeans) и выходом в Интернет.
3.	Групповые (индивидуальные) консультации	Аудитория, оснащенная персональными компьютерами с установленным программным обеспечением (JDK, NetBeans).
4.	Текущий контроль, промежуточная аттестация	Аудитория, оснащенная персональными компьютерами с установленным программным обеспечением (JDK, NetBeans).
5.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет»,программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.