

МИНОБРНАУКИ РОССИИ



Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный гуманитарный университет»
(ФГБОУ ВО «РГГУ»)

Факультет «Информационных систем и безопасности»
Кафедра «Информационных технологий и систем»

ПРОГРАММИРОВАНИЕ (JAVA)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Направление подготовки 09.03.03 «Прикладная информатика»
Направленность «Прикладная информатика в гуманитарной сфере»
Квалификация выпускника - бакалавр

Форма обучения - очная

РПД адаптирована для лиц
с ограниченными возможностями
здоровья и инвалидов

Москва 2021

ПРОГРАММИРОВАНИЕ (JAVA)

Рабочая программа дисциплины

Составитель:

к.х.н., доцент А.М. Подорожный

Ответственный редактор

к.т.н., доцент, заведующий кафедрой

информационных технологий и систем А.А. Роганов

УТВЕРЖДЕНО

Протокол заседания кафедры ИТиС

№ 12 от 28.06. 2021 г.

ОГЛАВЛЕНИЕ

ПРОГРАММИРОВАНИЕ (JAVA)	1
1 Пояснительная записка	4
1.1 Цель и задачи дисциплины	4
1.2 Формируемые компетенции, соотнесённые с планируемыми результатами обучения по дисциплине:.....	4
1.3 Место дисциплины в структуре образовательной программы	4
2 Структура дисциплины.....	5
3 Содержание дисциплины	6
4 Образовательные технологии	7
5 Оценка планируемых результатов обучения.....	10
5.1 Система оценивания	10
5.2 Критерии выставления оценки по дисциплине.....	10
5.3 Оценочные средства (материалы) для текущего контроля успеваемости, промежуточной аттестации обучающихся по дисциплине	12
6 Учебно-методическое и информационное обеспечение дисциплины.....	13
6.1 Список литературы	13
6.2 Перечень ресурсов информационно-коммуникационной среды “Интернет»	13
7 Материально-техническое обеспечение дисциплины	14
8 Обеспечение образовательного процесса для лиц с ограниченными возможностями здоровья и инвалидов	14
9 Методические материалы.....	16
9.1 Планы практических занятий	16
Приложения	
Приложение 1. Аннотация дисциплины.....	18

1 Пояснительная записка

1.1 Цель и задачи дисциплины

Цель дисциплины – приобретение знаний, навыков и умений в области программирования на языке высокого уровня Java.

Задачи дисциплины:

1. Изучение базовых принципов программирования на языке Java;
2. Дать широко применяемые технологии и методы программирования для разработки приложений на языке Java, в том числе, специализированные методы обработки и хранения данных;
3. Изучение главных управляющих структур языков Java;
4. Познакомить со стандартами языков Java и их отличительными особенностями;
5. Дать представления о процедуре отладки разрабатываемых приложений.

1.2 Формируемые компетенции, соотнесённые с планируемыми результатами обучения по дисциплине:

Компетенция	Индикаторы компетенций	Результаты обучения
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Знает основные языки программирования, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.	Знать: базовые принципы программирования на языке высокого уровня; главные управляющие структуры языка Java; принципы объектно-ориентированного программирования; алгоритмы отладки разрабатываемых приложений.
	ОПК-7.2. Умеет применять языки программирования, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.	Уметь: применять методы обработки и хранения данных языка Java, разрабатывать алгоритмы для решения задач последовательного и объектно-ориентированного программирования, в том числе для разработки информационных систем и автоматизации бизнес-процессов, разрабатывать модели реляционных баз данных и информационных хранилищ на языке высокого уровня Java
	ПК-7.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач	Владеть: навыками программирования, отладки и тестирования на языке высокого уровня Java, основными управляющими структурами языка Java; методами сортировки и хранения данных.

1.3 Место дисциплины в структуре образовательной программы

Дисциплина «Программирование (Java)» относится к базовой части блока учебного плана по направлению подготовки «Прикладная информатика», профиль «Прикладная

информатика в гуманитарной сфере». Имеет индекс Б1.О.13.03 в структуре учебного плана. Для освоения дисциплины необходимы знания, умения и владения, сформированные в ходе изучения следующих дисциплин: «Математический анализ», «Линейная алгебра и аналитическая геометрия», «Информационные технологии», «Информационные системы», «Программирование C++», «Программирование C#».

В результате освоения дисциплины формируются знания, умения и владения, необходимые для изучения следующих дисциплин: «Программная инженерия», «Администрирование информационных систем гуманитарной сферы», «Программирование интерфейсов в задачах информатизации гуманитарной сферы», «Проектный практикум в информатизации гуманитарной сферы».

2 Структура дисциплины

Структура дисциплины для очной формы обучения

Общая трудоёмкость дисциплины составляет 3 з.е., 114 ч., в том числе контактная работа обучающихся с преподавателем 42 ч., самостоятельная работа обучающихся 54 ч.

№ п/п	Раздел дисциплины/темы	Семестр	Виды учебной работы (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации
			контактная					
			Лекции	Семинар	Практические занятия	Промежуточная аттестация		
1	Раздел 1. Основы программирования на языке Java.	3	2		8		10	Защита отчета по практической работе №1
2	Раздел 2. Классы. Наследование. Многопоточное программирование (Java).	3	4		6		12	Защита отчета по практической работе №2
3	Раздел 3. Лямбда-функции. Введение в библиотеку Java.	3	2		6		8	Защита отчета по практической работе №3
4	Раздел 4. Чтение и запись данных в файлы. Автоматическое закрытие файла.	3	2		6		10	Защита отчета по практической работе №4
5	Раздел 5. Чтение и запись данных в файлы. Автоматическое закрытие файла. Основы создания апплетов.	3	2		4		14	Защита отчета по практической работе №5
	Экзамен	3				18		Экзамен по билетам
	итого:		12		30		54	

3 Содержание дисциплины

№	Наименование раздела дисциплины	Содержание
1	Основы программирования на языке Java.	Отличительные особенности Java. Объектно-ориентированное программирование. Примитивные типы. Числа с плавающей точкой. Символы. Логические значения. Переменные. Преобразование и приведение типов. Автоматическое продвижение типов в выражениях. Массивы. Арифметические операции. Поразрядные операции. Операции отношения. Операция присваивания. Тернарная операция ?. Операторы выбора. Условный оператор if . Оператор switch . Операторы цикла Цикл while . Цикл do...while . Цикл for . Вложенные циклы. Операторы перехода. Применение оператора break . Применение оператора continue . Оператор return .
2	Классы. Наследование. Многопоточное программирование (Java).	Основы классов. Общая форма класса. Простой класс. Объявление объектов. Подробное рассмотрение оператора new . Присваивание переменным ссылок на объекты. Введение в методы. Ввод метода в класс Box . Возврат значений. Ввод метода, принимающего параметры. Конструкторы. Параметризованные конструкторы. Ключевое слово this . Соккрытие переменной экземпляра. Сборка "мусора". Метод finalize() . Класс Stack . Перегрузка методов. Перегрузка конструкторов. Применение объектов в качестве параметров. Подробное рассмотрение особенностей передачи аргументов. Возврат объектов. Рекурсия. Введение в управление доступом. Ключевое слово static . Ключевое слово final . Основы наследования. Доступ к членам класса и наследование. Ключевое слово super . Создание многоуровневой иерархии. Порядок вызова конструкторов. Переопределение методов. Применение абстрактных классов. Модель потоков исполнения в Java . Главный поток исполнения. Создание потока исполнения. Создание многих потоков исполнения. Применение методов isAlive() и join() .
3	Лямбда-функции. Введение в библиотеку Java.	Введение в лямбда-выражения. Блочные лямбда-выражения. Обобщенные функциональные интерфейсы. Передача лямбда-выражений в качестве аргументов. Лямбда-выражения и исключения. Лямбда-выражения и захват переменных. Ссылки на методы. Ссылки на конструкторы. Конструкторы символьных строк. Длина символьной строки. Специальные

		строковые операции. Извлечение символов. Сравнение символьных строк. Поиск в символьных строках. Пакет java.lang . Пакет java.util .
4	Чтение и запись данных в файлы. Автоматическое закрытие файла.	Основы ввода-вывода. Поток ввода-вывода. Поток ввода-вывода байтов и символов. Предопределенные потоки ввода-вывода. Чтение данных, вводимых с консоли. Запись данных, выводимых на консоль. Класс PrintWriter . Чтение и запись данных в файлы. Автоматическое закрытие файла.
5	Чтение и запись данных в файлы. Автоматическое закрытие файла. Основы создания апплетов.	Основы создания апплетов. Модификаторы доступа transient и volatile . Применение оператора instanceof . Модификатор доступа strictfp . Платформенно-ориентированные методы. Трудности, связанные с платформенно-ориентированными методами. Применение ключевого слова assert . Параметры включения и отключения режима проверки утверждений. Статический импорт. Вызов перегружаемых конструкторов по ссылке this() . Компактные профили Java API . Классы системы ввода-вывода NIO. Основные положения о системе ввода-вывода NIO. Применение системы NIO для канального ввода-вывода. Применение системы NIO для потокового ввода-вывода. Применение системы ввода-вывода NIO для операций в файловой системе.

4 Образовательные технологии

№ п/п	Наименование раздела	Виды учебных занятий	Образовательные технологии
1	2	3	4
1.	Все разделы	Лекция.	Лекция с использованием иллюстраций и видеоматериалов (на проекторе), с показом образцов аппаратуры, печатных материалов и других изделий. Лекции имеют элементы интерактивности: допускается дискуссия, поощряются правильные ответы на вопросы. Лекционные занятия включают изложение, обсуждение и разъяснение основных направлений и вопросов изучаемой дисциплины, знание которых необходимо в ходе реализации всех остальных видов занятий и в самостоятельной работе студентов. На лекциях студенты получают самые необходимые знания по изучаемой проблеме. Непременным условием для глубокого и прочного усвоения учебного материала является умение студентов сосредоточенно слушать лекции, активно, творчески воспринимать излагаемые сведения. Внимательное слушание лекций

			<p>предполагает интенсивную умственную деятельность студента. Краткие записи лекций, конспектирование их помогает усвоить материал. Конспект является полезным тогда, когда записано самое существенное, основное. Запись лекций рекомендуется вести по возможности собственными формулировками. Желательно запись осуществлять на одной странице, а следующую оставлять для проработки учебного материала самостоятельно в домашних условиях. Конспект лучше подразделять на пункты, параграфы, соблюдая красную строку. Принципиальные места, определения, формулы следует сопровождать замечаниями.</p>
2.	Все разделы	Практическое занятие.	<p>Предлагается следующая последовательность освоения новых, незнакомых программ.</p> <p>После включения программы надо определить, где находятся ее основные настройки, при необходимости их произвести. Затем вызвать шаблон создаваемого программой документа (если его еще нет), освоить приемы форматирования и настройки характеристик шаблона. После этого – загрузить шаблон данными в виде вызванного файла, либо с помощью клавиатуры и мыши. Далее, манипулируя введенными данными, полезно познакомиться с важнейшими командами и инструментами программы. Как правило, они легкодоступны и находятся на рабочем столе.</p> <p>Необходимо также освоить другие команды, скрытые в меню, но применяемые для решения часто встречающихся на практике задач. Чем чаще встречается задача, тем лучше должны быть освоены способы ее решения. Здесь, правда, возможен элемент субъективизма, поскольку каждому в его деятельности могут встречаться различные задачи.</p> <p>Таким образом, логично осваивать новую программу в следующей последовательности: включение, базовые настройки, загрузка информации, основные инструменты, другие часто используемые команды.</p> <p>Развитое, конкурентоспособное на мировом рынке программное средство обычно содержит сотни и тысячи команд. Например, Microsoft Word содержит около тысячи команд. Запомнить и профессионально освоить их за ограниченное время невозможно, да и не нужно. Но все же полезно ознакомиться со спектром возможностей программы. Тогда если при работе перед вами встанет нестандартная задача, то можно вспомнить, что ее в принципе можно решить, хотя вряд ли удастся вспомнить, как это делается. Но если решение действительно необходимо, открывает новые возможности, или существенно экономит время и силы, то поэкспериментировав, почитав литературу, расспросив специалистов, можно в конце концов реализовать нужную функцию компьютерной</p>

			<p>программы. Главное – знать, что такая функция существует.</p> <p>То есть, полезно не только хорошо освоить базовые действия, но и максимально познакомиться с остальными возможностями.</p> <p>Освоить компьютерную программу можно только в ходе самостоятельной практической работы с реальными задачами, когда проявляются основные функции программы и элементарные пробелы в подготовке пользователя. Именно такие задачи взяты за основу практических занятий. При этом составлено по возможности подробное, безошибочное и ясное описание порядка действий по выполнению заданий.</p> <p>Во всех практических занятиях студентам предлагается выполнить конкретные практические задания. В большинстве заданий сначала надо загрузить исходные файлы с текстами, изображениями и пр. для последующей работы с ними. Исходные файлы являются неотъемлемой частью практикума. В некоторых работах имеются образцы правильного выполнения заданий, в виде иллюстраций форматов JPEG или TIFF.</p>
3.	Все разделы	Самостоятельная работа	<p>Целью самостоятельной (внеаудиторной) работы студентов является обучение навыкам работы с научно-теоретической, периодической, научно-технической литературой и технической документацией, необходимыми для углубленного изучения дисциплины «Программирование (Java)», а также развитие у них устойчивых способностей к самостоятельному изучению и изложению полученной информации.</p> <p>Основными задачами самостоятельной работы студентов являются:</p> <ul style="list-style-type: none"> – овладение фундаментальными знаниями; – наработка профессиональных навыков; – приобретение опыта творческой и исследовательской деятельности; – развитие творческой инициативы, самостоятельности и ответственности студентов.

5 Оценка планируемых результатов обучения

5.1 Система оценивания

Форма контроля	Макс. количество баллов	
	За одну работу	Всего
Текущий контроль: - защита отчета по практической работе	12 баллов	60 баллов
Промежуточная аттестация: экзамен		40 баллов
Итого за семестр		100 баллов

Полученный совокупный результат конвертируется в традиционную шкалу оценок и в шкалу оценок Европейской системы переноса и накопления кредитов (European Credit Transfer System; далее – ECTS) в соответствии с таблицей:

100-балльная шкала	Традиционная шкала		Шкала ECTS
95 – 100	отлично	зачтено	A
83 – 94			B
68 – 82	хорошо		C
56 – 67	удовлетворительно		D
50 – 55			E
20 – 49	неудовлетворительно	не зачтено	FX
0 – 19			F

5.2 Критерии выставления оценки по дисциплине

Баллы/ Шкала ECTS	Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
100-83/ A,B	«отлично»/ «зачтено (отлично)»/ «зачтено»	<p>Выставляется обучающемуся, если он глубоко и прочно усвоил теоретический и практический материал, может продемонстрировать это на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся исчерпывающе и логически стройно излагает учебный материал, умеет увязывать теорию с практикой, справляется с решением задач профессиональной направленности высокого уровня сложности, правильно обосновывает принятые решения.</p> <p>Свободно ориентируется в учебной и профессиональной литературе.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «высокий».</p>

Баллы/ Шкала ECTS	Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
82-68/ С	«хорошо»/ «зачтено (хорошо)»/ «зачтено»	<p>Выставляется обучающемуся, если он знает теоретический и практический материал, грамотно и по существу излагает его на занятиях и в ходе промежуточной аттестации, не допуская существенных неточностей.</p> <p>Обучающийся правильно применяет теоретические положения при решении практических задач профессиональной направленности разного уровня сложности, владеет необходимыми для этого навыками и приёмами.</p> <p>Достаточно хорошо ориентируется в учебной и профессиональной литературе.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «хороший».</p>
67-50/ D,E	«удовлетворительно»/ «зачтено (удовлетворительно)»/ «зачтено»	<p>Выставляется обучающемуся, если он знает на базовом уровне теоретический и практический материал, допускает отдельные ошибки при его изложении на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся испытывает определённые затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, владеет необходимыми для этого базовыми навыками и приёмами.</p> <p>Демонстрирует достаточный уровень знания учебной литературы по дисциплине.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «достаточный».</p>
49-0/ F,FX	«неудовлетворительно»/ не зачтено	<p>Выставляется обучающемуся, если он не знает на базовом уровне теоретический и практический материал, допускает грубые ошибки при его изложении на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся испытывает серьёзные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приёмами.</p> <p>Демонстрирует фрагментарные знания учебной литературы по дисциплине.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p>

Баллы/ Шкала ECTS	Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
		Компетенции на уровне «достаточный», закреплённые за дисциплиной, не сформированы.

5.3 Оценочные средства (материалы) для текущего контроля успеваемости, промежуточной аттестации обучающихся по дисциплине

Вопросы для подготовки к экзамену

1. Отличительные особенности Java. (ОПК-7.1)
2. Объектно-ориентированное программирование. (ОПК-7.1)
3. Примитивные типы. Числа с плавающей точкой. Символы. (ОПК-7.3)
4. Логические значения. Переменные. Преобразование и приведение типов. Автоматическое продвижение типов в выражениях. (ОПК-7.3)
5. Массивы. Арифметические операции. Поразрядные операции. Операции отношения. (ОПК-7.1)
6. Операция присваивания. Тернарная операция. Операторы выбора. (ОПК-7.2)
7. Условный оператор if. Оператор switch. (ОПК-7.2)
8. Операторы цикла. Цикл while. Цикл do...while. Цикл for. (ОПК-7.2)
9. Вложенные циклы. Операторы перехода. Применение оператора break. (ОПК-7.3)
10. Применение оператора continue. Оператор return. (ОПК-7.3)
11. Основы классов. Общая форма класса. Простой класс. Объявление объектов. (ОПК-7.1)
12. Подробное рассмотрение оператора new. Присваивание переменным ссылок на объекты. (ОПК-7.1)
13. Введение в методы. Ввод метода в класс Vox. Возврат значений. Ввод метода, принимающего параметры. (ОПК-7.1)
14. Конструкторы. Параметризованные конструкторы. (ОПК-7.1)
15. Ключевое слово this. Соккрытие переменной экземпляра. Сборка "мусора". Метод finalize(). (ОПК-7.3)
16. Класс Stack. Перегрузка методов. Перегрузка конструкторов. (ОПК-7.2)
17. Применение объектов в качестве параметров. (ОПК-7.2)
18. Возврат объектов. Рекурсия. Введение в управление доступом. (ОПК-7.3)
19. Ключевое слово static. Ключевое слово final. (ОПК-7.2)
20. Основы наследования. Доступ к членам класса и наследование. (ОПК-7.1)
21. Ключевое слово super. Создание многоуровневой иерархии. (ОПК-7.2)
22. Порядок вызова конструкторов. Переопределение методов. (ОПК-7.2)
23. Применение абстрактных классов. (ОПК-7.1)
24. Модель потоков исполнения в Java. Главный поток исполнения. (ОПК-7.3)
25. Создание потока исполнения. Создание многих потоков исполнения. (ОПК-7.3)
26. Применение методов isAlive() и join(). (ОПК-7.3)
27. Введение в лямбда-выражения. Блочные лямбда-выражения. (ОПК-7.1)
28. Обобщенные функциональные интерфейсы. Передача лямбда-выражений в качестве аргументов. (ОПК-7.2)
29. Лямбда-выражения и исключения. Лямбда-выражения и захват переменных. (ОПК-7.2)
30. Ссылки на методы. Ссылки на конструкторы. (ОПК-7.2)
31. Конструкторы символьных строк. Длина символьной строки. (ОПК-7.1)
32. Специальные строковые операции. Извлечение символов. (ОПК-7.3)
33. Сравнение символьных строк. Поиск в символьных строках. (ОПК-7.3)
34. Пакет java.lang. Пакет java.util. (ОПК-7.3)

35. Основы ввода-вывода. Потоки ввода-вывода. (ОПК-7.1)
36. Потоки ввода-вывода байтов и символов. Предопределенные потоки ввода-вывода. (ОПК-7.2)
37. Чтение данных, вводимых с консоли. Запись данных, выводимых на консоль. (ОПК-7.1)
38. Класс PrintWriter. Чтение и запись данных в файлы. (ОПК-7.2)
39. Автоматическое закрытие файла. (ОПК-7.2)
40. Основы создания апплетов. Модификаторы доступа transient и volatile. (ОПК-7.1)
41. Применение оператора instanceof. Модификатор доступа strictfp. (ОПК-7.3)
42. Платформенно-ориентированные методы. (ОПК-7.1)
43. Трудности, связанные с платформенно-ориентированными методами. (ОПК-7.3)
44. Применение ключевого слова assert. (ОПК-7.3)
45. Параметры включения и отключения режима проверки утверждений. (ОПК-7.2)
46. Статический импорт. Вызов перегружаемых конструкторов по ссылке this(). (ОПК-7.1)
47. Компактные профили Java API. Классы системы ввода-вывода NIO. (ОПК-7.2)
48. Основные положения о системе ввода-вывода NIO. (ОПК-7.1)
49. Применение системы NIO для канального и потокового ввода-вывода. (ОПК-7.2)
50. Применение системы ввода-вывода NIO для операций в файловой системе. (ОПК-7.2)

6 Учебно-методическое и информационное обеспечение дисциплины

6.1 Список литературы

Основная

1. Пруцков, А.В. Программирование на языке Java. Введение в курс с примерами и практическими заданиями : учебник. — Москва : КУРС, 2018.- 208 с. - ISBN 978-5-906923-51-6. – Режим доступа: <https://new.znanium.com/catalog/product/1017180>.
2. Гуськова, О.И. Объектно ориентированное программирование в Java : учебное пособие. - Москва : МПГУ, 2018. - 240 с. - ISBN 978-5-4263-0648-6. - Режим доступа: <https://new.znanium.com/catalog/product/1020593>.
3. С. Стоянович, А. Симович. Бессерверные приложения на JavaScript : практическое руководство / пер. с англ. А. Н. Киселева. - Москва : ДМК Пресс, 2020. - 394 с. - ISBN 978-5-97060-782-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1094948>

Дополнительная

1. Васюткина, И. А. Технология разработки объектно-ориентированных программ на JAVA. - Новосибирск :НГТУ, 2012. - 152 с.: ISBN 978-5-7782-1973-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/557111>
2. Риз, Р. Обработка естественного языка на Java / Р. Риз ; пер. с англ. А.В. Снастина. - Москва : ДМК Пресс, 2016. - 264 с. - ISBN 978-5-97060-331-4. - Текст : электронный. - URL: <http://znanium.com/catalog/product/1028050>
3. Хеффельфингер, Д. Разработка приложений Java EE 7 в NetBeans 8 / Дэвид Хеффельфингер ; пер. с англ. А.Н. Киселева. - Москва : ДМК Пресс, 2016. - 348 с. - ISBN 978-5-97060-329-1. - Текст : электронный. - URL: <http://znanium.com/catalog/product/1028113>

6.2 Перечень ресурсов информационно-коммуникационной среды «Интернет»

1. Официальный сайт. <https://www.java.com/ru/>

2. Руководство по началу работы с Java в среде IDE NetBeans
https://netbeans.org/kb/71/java/quickstart_ru.html
3. <http://www.consultant.ru/> - КонсультантПлюс компьютерная справочная правовая система.
4. <https://www.scopus.com> – Международная реферативная наукометрическая БД, Scopus, доступна по подписке.

7 Материально-техническое обеспечение дисциплины

Для реализации программы дисциплины требуется:

- компьютеры, с предустановленным базовым программным обеспечением в составе ОС Windows 10 и MS Office;
- объединение компьютеров в локальную сеть компьютерного класса с высокоскоростным выходом в интернет;
- для лекционного курса – посадочные места по количеству обучающихся, доска, мультимедиа проектор с экраном;
- рабочее место преподавателя, оснащенное компьютером с выходом на проектор.

Используемое программное обеспечение:

Наименование ПО	Лицензия/сертификат/заказ	Дата лицензии
Windows 10	68526624	без даты
Среда разработки NetBeans IDE 8.2	свободный доступ	свободный доступ
Виртуальная машина JRE 8	свободный доступ	свободный доступ
Microsoft Office 2010 Pro	49420326	08.12.2011
Mozilla Firefox 52.8.1 ESR	свободный доступ	свободный доступ
Kaspersky Endpoint Security	11501912170602202 301550	17.12.2019
Платформа ZOOM	лицензионное	

8 Обеспечение образовательного процесса для лиц с ограниченными возможностями здоровья и инвалидов

В ходе реализации дисциплины используются следующие дополнительные методы обучения, текущего контроля успеваемости и промежуточной аттестации обучающихся в зависимости от их индивидуальных особенностей:

- для слепых и слабовидящих:
 - лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
 - письменные задания выполняются на компьютере со специализированным программным обеспечением, или могут быть заменены устным ответом;
 - обеспечивается индивидуальное равномерное освещение не менее 300 люкс;
 - для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств;
 - письменные задания оформляются увеличенным шрифтом;
 - экзамен и зачет проводятся в устной форме или выполняются в письменной форме на компьютере.
- для глухих и слабослышащих:

- лекции оформляются в виде электронного документа, либо предоставляется звукоусиливающая аппаратура индивидуального пользования;
- письменные задания выполняются на компьютере в письменной форме;
- экзамен и зачёт проводятся в письменной форме на компьютере; возможно проведение в форме тестирования.

- для лиц с нарушениями опорно-двигательного аппарата:
 - лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
 - письменные задания выполняются на компьютере со специализированным программным обеспечением;
 - экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере.

При необходимости предусматривается увеличение времени для подготовки ответа.

Процедура проведения промежуточной аттестации для обучающихся устанавливается с учётом их индивидуальных психофизических особенностей. Промежуточная аттестация может проводиться в несколько этапов.

При проведении процедуры оценивания результатов обучения предусматривается использование технических средств, необходимых в связи с индивидуальными особенностями обучающихся. Эти средства могут быть предоставлены университетом, или могут использоваться собственные технические средства.

Проведение процедуры оценивания результатов обучения допускается с использованием дистанционных образовательных технологий.

Обеспечивается доступ к информационным и библиографическим ресурсам в сети Интернет для каждого обучающегося в формах, адаптированных к ограничениям их здоровья и восприятия информации:

- для слепых и слабовидящих:
 - в печатной форме увеличенным шрифтом;
 - в форме электронного документа;
 - в форме аудиофайла.
- для глухих и слабослышащих:
 - в печатной форме;
 - в форме электронного документа.
- для обучающихся с нарушениями опорно-двигательного аппарата:
 - в печатной форме;
 - в форме электронного документа;
 - в форме аудиофайла.

Учебные аудитории для всех видов контактной и самостоятельной работы, научная библиотека и иные помещения для обучения оснащены специальным оборудованием и учебными местами с техническими средствами обучения:

- для слепых и слабовидящих:
 - устройством для сканирования и чтения с камерой SARA CE;
 - дисплеем Брайля PAC Mate 20;
 - принтером Брайля EmBraille ViewPlus;
- для глухих и слабослышащих:
 - автоматизированным рабочим местом для людей с нарушением слуха и слабослышащих;
 - акустический усилитель и колонки;
- для обучающихся с нарушениями опорно-двигательного аппарата:
 - передвижными, регулируемые эргономическими партами СИ-1;
 - компьютерной техникой со специальным программным обеспечением.

9 Методические материалы

9.1 Планы практических занятий

Тема 1 (4 ч.). Основы программирования на языке Java .

Задание:

1. Согласно заранее определенному варианту выполнить программирование операторов цикла и ветвления для решения алгебраической задачи;
2. Типизацию переменных задачи осуществить согласно типам данных Java;
3. Подготовить развернутый отчет о проделанной работе.

Тема 2 (4 ч.). Классы. Наследование. Многопоточное программирование (Java).

Задание:

1. Согласно заранее определенному варианту выполнить объявление многомерного статического массива;
2. Выполнить поиск и перестановку минимального и максимального элемента.
3. Подготовить развернутый отчет о проделанной работе.

Тема 3 (4 ч.). Лямбда-функции. Введение в библиотеку Java.

Задание:

1. Согласно заранее определенному варианту выполнить программирование алгоритмов сортировки с использованием лямбда-функции Java;
2. Методы сортировки должны быть реализованы с использованием библиотек `java.lang` ;
3. Подготовить развернутый отчет о проделанной работе.

Тема 4 (4 ч.). Чтение и запись данных в файлы. Автоматическое закрытие файла.

Задание:

1. Согласно заранее определенному варианту выполнить обработку C-подобных строковых данных и осуществить чтение/запись в файл;
2. Предусмотреть обработку текстовых(числовых) данных, хранящихся в файле, при помощи инструмента **struct**;
3. Подготовить развернутый отчет о проделанной работе.

Тема 5 (4 ч.). Чтение и запись данных в файлы. Автоматическое закрытие файла. Основы создания апплетов.

Задание:

1. Согласно заранее определенному варианту и используя принципы объектно-ориентированного программирования организовать создание объекта типа **class** в состав, которого входят методы сортировки динамически инициализированного двумерного массива, а также определены конструктор и деструктор класса;
2. Предусмотреть возможность создания дочерних классов на базе созданного родительского класса;
3. Подготовить развернутый отчет о проделанной работе.

9.2 Методические рекомендации по подготовке письменных работ

Отчет по проделанной работе должен быть изложен с соблюдением правил грамматики русского и английского языков (в случаях необходимости). При этом отражаемые результаты работы должны быть информативными, тезисного порядка. В отчет входят следующие обязательные разделы:

1. Титульный лист с полным указанием ведомственной принадлежности, названия ВУЗа, института, факультета, кафедры. Кроме того, полное точное название лабораторной работы, Ф.И.О. студента, подготовившего отчет о результатах проделанной работы и Ф.И.О., должность, название кафедры преподавателя осуществляющего проверку и оценивание полученных результатов.
2. Содержание.
3. Введение.
4. Цели и задачи практической работы.
5. Методы и технологии, применяемые для решения поставленных задач оформленные в виде отдельных этапов работы.
6. Выводы по работе.
7. Приложения.

Оформление отчета выполняется с использованием компьютерной верстки **LaTeX**. Отчет сохраняется и представляет для проверки в виде отдельного **pdf** файла. В имени файла указывается фамилия студента и номер выполненной работы.

АННОТАЦИЯ ДИСЦИПЛИНЫ «ПРОГРАММИРОВАНИЕ (JAVA)»

Дисциплина реализуется на факультете ФИСБ, кафедрой Информационных технологий и систем.

Цель дисциплины – приобретение знаний, навыков и умений в области программирования на языке высокого уровня Java.

Задачи дисциплины:

1. Изучение базовых принципов программирования на языке Java;
2. Дать широко применяемые технологии и методы программирования для разработки приложений на языке Java, в том числе, специализированные методы обработки и хранения данных;
3. Изучение главных управляющих структур языков Java;
4. Изучение принципов объектно-ориентированного программирования. Понятие о классе и его свойствах;
5. Сформировать навыки и умения по разработке алгоритмов в задачах последовательного и объектно-ориентированного программирования;
6. Познакомить со стандартами языков Java и их отличительными особенностями;
7. Дать представления о процедуре отладки разрабатываемых приложений;

Дисциплина направлена на формирование следующих компетенций.

- ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения
 - ОПК-7.1. Знает основные языки программирования, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.
 - ОПК-7.2. Умеет применять языки программирования, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.
 - ОПК-7.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.

В результате освоения дисциплины обучающийся должен:

Знать: базовые принципы программирования на языке высокого уровня; главные управляющие структуры языка Java; принципы объектно-ориентированного программирования; алгоритмы отладки разрабатываемых приложений.

Уметь: применять методы обработки и хранения данных языка Java, разрабатывать алгоритмы для решения задач последовательного и объектно-ориентированного программирования, в том числе для разработки информационных систем и автоматизации бизнес-процессов, разрабатывать модели реляционных баз данных и информационных хранилищ на языке высокого уровня Java.

Владеть: навыками программирования, отладки и тестирования на языке высокого уровня Java, основными управляющими структурами языка Java; методами сортировки и хранения данных.

По дисциплине предусмотрена промежуточная аттестация в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 3 зачетных единицы.