

МИНОБРНАУКИ РОССИИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Рабочая программа дисциплины (модуля)
ПРОГРАММИРОВАНИЕ НА JAVA

Направление и направленность (профиль)
09.03.02 Информационные системы и технологии. Информационные системы и технологии

Год набора на ОПОП
2020

Форма обучения
заочная

Владивосток 2024

Рабочая программа дисциплины (модуля) «Программирование на Java» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии (утв. приказом Минобрнауки России от 19.09.2017г. №926) и Порядком организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры (утв. приказом Минобрнауки России от 06.04.2021 г. N245).

Составитель(и):

Гриняк В.М., доктор технических наук, профессор, Кафедра информационных технологий и систем, Viktor.Grinyak@vvsu.ru

Утверждена на заседании кафедры информационных технологий и систем от 29.05.2024 , протокол № 9

СОГЛАСОВАНО:

Заведующий кафедрой (разработчика)

Кийкова Е.В.

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ	
Сертификат	1575633692
Номер транзакции	0000000000CEAE5A
Владелец	Кийкова Е.В.

1 Цель, планируемые результаты обучения по дисциплине (модулю)

Целью изучения дисциплины «Программирование на Java» является усвоение и закрепление основных приемов, методов и принципов работы при создании кроссплатформенных программ, усвоение навыков использования языка Java, подготовка к профессиональной сертификации.

Основные задачи изучения дисциплины:

- развитие навыков объектно-ориентированного программирования;
- освоение подходов к созданию консольных и визуальных кроссплатформенных программ;
- знакомство с понятиями и языком предметной области, в том числе международной англоязычной терминологией;
- развитие навыков работы в коллективе;
- подготовка к профессиональной сертификации Java-программиста.

Планируемыми результатами обучения по дисциплине (модулю), являются знания, умения, навыки. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы, представлен в таблице 1.

Таблица 1 – Компетенции, формируемые в результате изучения дисциплины (модуля)

Название ОПОП ВО, сокращенное	Код и формулировка компетенции	Код и формулировка индикатора достижения компетенции	Результаты обучения по дисциплине		
			Код результата	Формулировка результата	
09.03.02 «Информационные системы и технологии» (Б-ИС)	ПКВ-1 : Способен осуществлять интеграцию и тестирование работоспособности программных модулей и компонент программного обеспечения	ПКВ-1.2к : Проводит тестирование компонентов программного обеспечения ИС	РД2	Умение	создавать и тестировать программы на Java, в том числе для решения профессиональных задач с использованием возможностей инструментария среды программирования
			РД1	Знание	основных особенностей языка, принципов и приемов программирования, инструментальных средств программирования на Java, классификации платформ Java и возможностей их применения
					РД3

2 Место дисциплины (модуля) в структуре ОПОП

Отнесение дисциплины к элективным дисциплинам учебного плана определяется спецификой и миссией ВВГУ, а также особенностями взаимодействия ВВГУ с рынком труда и региональными требованиями, выраженными в результатах образования и компетенциях.

Входными требованиями, необходимыми для освоения дисциплины, является наличие у обучающихся компетенций, сформированных при изучении дисциплин «Информатика и основы программирования», «Программирование на С», «Технология программирования».

3. Объем дисциплины (модуля)

Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу с обучающимися (по видам учебных занятий) и на самостоятельную работу, приведен в таблице 2.

Таблица 2 – Общая трудоемкость дисциплины

Название ОПОП ВО	Форма обучения	Часть УП	Семестр (ОФО) или курс (ЗФО, ОЗФО)	Трудо-емкость (З.Е.)	Объем контактной работы (час)					СРС	Форма аттестации	
					Всего	Аудиторная			Внеаудиторная			
						лек.	прак.	лаб.	ПА			КСР
09.03.02 Информационные системы и технологии	ЗФО	Б1.ДВ.Б	3	3	13	4	8	0	1	0	95	ДЗ

4 Структура и содержание дисциплины (модуля)

4.1 Структура дисциплины (модуля) для ЗФО

Тематический план, отражающий содержание дисциплины (перечень разделов и тем), структурированное по видам учебных занятий с указанием их объемов в соответствии с учебным планом, приведен в таблице 3.1

Таблица 3.1 – Разделы дисциплины (модуля), виды учебной деятельности и формы текущего контроля для ЗФО

№	Название темы	Код результата обучения	Кол-во часов, отведенное на				Форма текущего контроля
			Лек	Практ	Лаб	СРС	
1	Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.	РД1, РД2, РД3	1	2	0	23	текущий опрос, отчет о выполнении практической работы
2	Разработка классов в языке Java. Наследование и интерфейсы в языке Java. Обобщённые типы и коллекции значений в языке Java. Работа со строками в языке Java.	РД1, РД2, РД3	1	2	0	24	текущий опрос, отчет о выполнении практической работы
3	Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.	РД1, РД2, РД3	1	2	0	24	текущий опрос, отчет о выполнении практической работы
4	Многопоточные программы Java. Параллельное программирование Java. Построение приложений баз данных с использованием JDBC API. Локализация Java программ.	РД1, РД2, РД3	1	2	0	24	текущий опрос, отчет о выполнении практической работы
Итого по таблице			4	8	0	95	

4.2 Содержание разделов и тем дисциплины (модуля) для ЗФО

Тема 1 Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.

Содержание темы: Структура классов Java. Программные блоки и комментарии. Переменные. Условные операторы и циклы. Инкапсуляция при разработке классов Java. Моделирование задачи с использованием классов Java. Неизменяемые классы. Подклассы: создание и использование. Перегрузка методов класса. Методы с переменным числом аргументов. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, практическое занятие.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, практическим занятиям.

Тема 2 Разработка классов в языке Java. Наследование и интерфейсы в языке Java. Обобщённые типы и коллекции значений в языке Java. Работа со строками в языке Java.

Содержание темы: Спецификаторы доступа private, protected, default и public. Перегрузка конструкторов и других методов. Использование оператора instanceof для определения типа объекта. Виртуальный вызов методов класса. Преобразование типов «вверх» (апкостинг) и «вниз» (даункостинг). Перегрузка методов класса Object. Использование абстрактных классов. Ключевые слова final и static. Шаблон проектирования singleton. Вложенные классы. Интерфейсы в Java, определение интерфейсов. Особенности использования интерфейсов и классов в программах. Расширение интерфейсов. Рефакторинг кода. Обобщённые типы как способ создания классов в Java. Создание объектов в рамках обобщённого типа. Создание коллекций без использования обобщённых типов и с их использованием. Работа со структурами данных ArrayList, Set, HashMap. Реализация стека и очереди. Перечислимые типы. Чтение данных из командной строки. Поиск строк. Парсинг строк. Создание строк с использованием класса StringBuilder. Поиск в строке, парсинг строки и удаление строк с использованием регулярных выражений. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, практическое занятие.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, практическим занятиям.

Тема 3 Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.

Содержание темы: Типы исключений в Java. Использование конструкций try и throw. Использование catch, единожды и многократно. Ключевое слово finally. Классы исключений. Создание выборочных исключений и автозакрываемых ресурсов. Использование assertions. Основы ввода и вывода в Java программах. Чтение данных с консоли и вывод данных на консоль. Использование потоков для чтения и записи файлов. Чтение и запись объектов с использованием сериализации. Использование интерфейса Path для работы с файлами. Работа с классом Files для операций над файлами. Канальный и потоковый ввод-вывод в файлах. Работа с атрибутами файлов. Доступ к дереву каталогов. Поиск файлов с использованием класса PathMatcher. .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, практическое занятие.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, практическим занятиям.

Тема 4 Многопоточные программы Java. Параллельное программирование Java. Построение приложений баз данных с использованием JDBC API. Локализация Java программ.

Содержание темы: Определение и создание потоков. Управление потоками. Синхронизация потоков. Проблемы многопоточного программирования. Атомарные переменные. Метод `ReentrantReadWriteLock()`. Работа с коллекцией `java.util.concurrent`. Синхронизирующие классы. Использование `ExecutorService`. Fork-Join фреймворк. Основные функции JDBC API. Подключение к базе данных с использованием драйвера JDBC. Подача запросов получение результатов из базы данных. Транзакции и JDBC. Использование паттерна `Data Access Object`. Особенности и задачи локализации программ. Определение и представление локализуемых данных. Чтение и установка локализуемых данных с помощью объекта `Locale`. Построение ресурсов. Вызов ресурсов из приложений. Форматирование текста и его локализация с использованием `NumberFormat` `DateFormat` .

Формы и методы проведения занятий по теме, применяемые образовательные технологии: лекция, практическое занятие.

Виды самостоятельной подготовки студентов по теме: подготовка к промежуточному тестированию, практическим занятиям.

5 Методические указания для обучающихся по изучению и реализации дисциплины (модуля)

5.1 Методические рекомендации обучающимся по изучению дисциплины и по обеспечению самостоятельной работы

В ходе изучения дисциплины «Программирование на Java» студенты могут посещать аудиторные занятия (лекции, практические занятия, консультации).

Особое место в овладении частью тем данной дисциплины может отводиться самостоятельной работе, при этом во время аудиторных занятий могут быть рассмотрены и проработаны наиболее важные и трудные вопросы по той или иной теме дисциплины, а второстепенные и более легкие вопросы, а также вопросы, специфичные для ОПОП, могут быть изучены студентами самостоятельно.

В соответствии с учебным планом процесс изучения дисциплины может предусматривать проведение лекций, практических занятий, консультаций, а также самостоятельную работу студентов. Обязательным является проведение практических занятий в специализированных компьютерных аудиториях, оснащенных подключенными к центральному серверу терминалами или персональными компьютерами.

В рамках общего объема часов, отведенных для изучения дисциплины, предусматривается выполнение следующих видов самостоятельных работ студентов: подготовка к практическим занятиям, работа над рекомендованной литературой, самостоятельное решение задач.

Практикуется интерактивный способ работы над задачами: студенты размещают решение задачи на форуме, указанном преподавателем, проходит обсуждение задачи и её решения, после чего преподаватель даёт свою оценку всем участникам процесса.

По тематике дисциплины выпущено большое количество печатных изданий. Электронные ресурсы могут быть использованы как источники справочной и обзорной информации по тематике дисциплины.

5.2 Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

При необходимости обучающимся из числа лиц с ограниченными возможностями здоровья и инвалидов (по заявлению обучающегося) предоставляется учебная информация в доступных формах с учетом их индивидуальных психофизических особенностей:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; индивидуальные консультации с привлечением

тифлосурдопереводчика; индивидуальные задания, консультации и др.

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания, консультации и др.

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; индивидуальные задания, консультации и др.

6 Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)

В соответствии с требованиями ФГОС ВО для аттестации обучающихся на соответствие их персональных достижений планируемым результатам обучения по дисциплине (модулю) созданы фонды оценочных средств. Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 1.

7 Учебно-методическое и информационное обеспечение дисциплины (модуля)

7.1 Основная литература

1. Васюткина, И. А. Разработка серверной части web-приложений на Java : учебное пособие / И. А. Васюткина. — Новосибирск : НГТУ, 2021. — 83 с. — ISBN 978-5-7782-4394-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/216155> (дата обращения: 18.07.2024). — Режим доступа: для авториз. пользователей.

2. Пономарчук, Ю. В. Программирование на языке Java : учебное пособие / Ю. В. Пономарчук, И. В. Кузнецов. — Хабаровск : ДВГУПС, 2021. — 103 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/259451> (дата обращения: 18.07.2024). — Режим доступа: для авториз. пользователей.

7.2 Дополнительная литература

1. Акчурин Э. А. Программирование на языке Java [Электронный ресурс] , 2011 - 317 - Режим доступа: <https://lib.rucont.ru/efd/319599>

2. Васюткина, И. А. Разработка клиент-серверных приложений на языке C# : учебное пособие / И. А. Васюткина. — Новосибирск : НГТУ, 2016. — 112 с. — ISBN 978-5-7782-2932-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/118210> (дата обращения: 18.07.2024). — Режим доступа: для авториз. пользователей.

3. Практикум по программированию на языке Java / А.Л. Мархакшинов, Н.Н. Шадрин .— Улан-Удэ : Бурятский государственный университет, 2017 .— 70 с. — ISBN 978-5-9793-0016-0 .— URL: <https://lib.rucont.ru/efd/609706> (дата обращения: 18.07.2024)

4. Программирование на JAVA [Электронный ресурс] , 2012 - 53 - Режим доступа: <https://lib.rucont.ru/efd/238627>

7.3 Ресурсы информационно-телекоммуникационной сети "Интернет", включая профессиональные базы данных и информационно-справочные системы (при необходимости):

1. IBM developerWorks Россия: Технология Java. Режим доступа:

<http://www.ibm.com/developerworks/ru/java/>

2. Java™ Platform, Standard Edition 7 API Specification. Режим доступа: <http://docs.oracle.com/javase/7/docs/api/index.html>

3. The Java Tutorials. Режим доступа: <http://docs.oracle.com/javase/tutorial/index.html>

4. Васюткина И. А. Технология разработки объектно-ориентированных программ на JAVA : учеб.- метод. пособие / И.А. Васюткина. – Новосибирск : Изд-во НГТУ, 2012. – 152 с. Режим доступа: <http://znanium.com/go.php?id=557111>

5. Электронная библиотечная система «РУКОНТ» - Режим доступа: <https://lib.rucont.ru/>

6. Электронно-библиотечная система "ЛАНЬ"

7. Электронно-библиотечная система "РУКОНТ"

8. Open Academic Journals Index (ОАИ). Профессиональная база данных - Режим доступа: <http://oaji.net/>

9. Президентская библиотека им. Б.Н.Ельцина (база данных различных профессиональных областей) - Режим доступа: <https://www.prlib.ru/>

10. Информационно-справочная система "Консультант Плюс" - Режим доступа: <http://www.consultant.ru/>

8 Материально-техническое обеспечение дисциплины (модуля) и перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения

Основное оборудование:

- Компьютеры
- Проектор
- Экран Projecta 160*160

Программное обеспечение:

- Microsoft Windows 7 Russian

МИНОБРНАУКИ РОССИИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Фонд оценочных средств
для проведения текущего контроля
и промежуточной аттестации по дисциплине (модулю)

ПРОГРАММИРОВАНИЕ НА JAVA

Направление и направленность (профиль)

09.03.02 Информационные системы и технологии. Информационные системы и технологии

Год набора на ОПОП
2020

Форма обучения
заочная

Владивосток 2024

1 Перечень формируемых компетенций

Название ОПОП ВО, сокращенное	Код и формулировка компетенции	Код и формулировка индикатора достижения компетенции
09.03.02 «Информационные системы и технологии» (Б-ИС)	ПКВ-1 : Способен осуществлять интеграцию и тестирование работ оспособности программных модулей и компонент программного обеспечения	ПКВ-1.2к : Проводит тестирование компонентов программного обеспечения ИС
	ПКВ-2 : Способен выполнять работы по созданию и сопровождению информационных систем	ПКВ-2.1к : Выполняет разработку прикладного программного обеспечения

Компетенция считается сформированной на данном этапе в случае, если полученные результаты обучения по дисциплине оценены положительно (диапазон критериев оценивания результатов обучения «зачтено», «удовлетворительно», «хорошо», «отлично»). В случае отсутствия положительной оценки компетенция на данном этапе считается несформированной.

2 Показатели оценивания планируемых результатов обучения

Компетенция ПКВ-1 «Способен осуществлять интеграцию и тестирование работоспособности программных модулей и компонент программного обеспечения»

Таблица 2.1 – Критерии оценки индикаторов достижения компетенции

Код и формулировка индикатора достижения компетенции	Результаты обучения по дисциплине			Критерии оценивания результатов обучения
	Код ре-з-та	Т и п ре з-та	Результат	
ПКВ-1.2к : Проводит тестирование компонентов программного обеспечения ИС	РД2	Умение	создавать и тестировать программы на Java, в том числе для решения профессиональных задач с использованием возможностей инструментария среды программирования	умение создавать и тестировать программы на Java

Компетенция ПКВ-2 «Способен выполнять работы по созданию и сопровождению информационных систем»

Таблица 2.2 – Критерии оценки индикаторов достижения компетенции

Код и формулировка индикатора достижения компетенции	Результаты обучения по дисциплине			Критерии оценивания результатов обучения
	Код ре-з-та	Т и п ре з-та	Результат	

ПКВ-2.1к : Выполняет разработку прикладного программного обеспечения	РД1	Знание	основных особенностей языка, принципов и приемов программирования, инструментальных средств программирования на Java, классификации платформ Java и возможностей их применения	достаточное для самостоятельного создания программ знание особенностей языка, приемов программирования, инструментальных средств программирования на Java
	РД3	Навык	использования приемов программирования, инструментария среды программирования Java и способов организации программ в том числе для решения прикладных задач	навыки создания программ на Java, которые могут быть использованы для решения прикладных задач

Таблица заполняется в соответствии с разделом 1 Рабочей программы дисциплины (модуля).

3 Перечень оценочных средств

Таблица 3 – Перечень оценочных средств по дисциплине (модулю)

Контролируемые планируемые результаты обучения	Контролируемые темы дисциплины	Наименование оценочного средства и представление его в ФОС		
		Текущий контроль	Промежуточная аттестация	
Заочная форма обучения				
РД1	Знание : основных особенностей языка, принципов и приемов программирования, инструментальных средств программирования на Java, классификации платформ Java и возможностей их применения	1.1. Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.	Практическая работа	Тест
		1.2. Разработка классов в языке Java. Наследование и интерфейсы в языке Java. Обобщённые типы и коллекции значений в языке Java. Работа с строками в языке Java.	Практическая работа	Тест
		1.3. Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.	Практическая работа	Тест
		1.4. Многопоточные программы Java. Параллельное программирование Java. Построение приложений баз данных с использованием JDBC API. Локализация Java программ.	Практическая работа	Тест
РД2	Умение : создавать и тестировать программы на Java, в том числе для решения профессиональных задач с использованием возможностей инструментария среды програм	1.1. Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.	Практическая работа	Тест

	мирования	1.2. Разработка классов в языке Java. Наследование и интерфейсы в языке Java. Обобщённые типы и коллекции значений в языке Java. Работа с строками в языке Java.	Практическая работа	Тест
		1.3. Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.	Практическая работа	Тест
		1.4. Многопоточные программы Java. Параллельное программирование Java. Построение приложений баз данных с использованием JDBC API. Локализация Java программ.	Практическая работа	Тест
РДЗ	Навык : использования приемов программирования, инструментария среды программирования Java и способов организации программ в том числе для решения прикладных задач	1.1. Синтаксис языка Java, классы в языке Java. Наследование и инкапсуляция в языке Java.	Практическая работа	Тест
		1.2. Разработка классов в языке Java. Наследование и интерфейсы в языке Java. Обобщённые типы и коллекции значений в языке Java. Работа с строками в языке Java.	Практическая работа	Тест
		1.3. Обработка исключений. Ввод и вывод в Java программах. Файловый ввод и вывод.	Практическая работа	Тест
		1.4. Многопоточные программы Java. Параллельное программирование Java. Построение приложений баз данных с использованием JDBC API. Локализация Java программ.	Практическая работа	Тест

4 Описание процедуры оценивания

Качество сформированности компетенций на данном этапе оценивается по результатам текущих и промежуточных аттестаций при помощи количественной оценки, выраженной в баллах. Максимальная сумма баллов по дисциплине (модулю) равна 100 баллам.

Вид учебной деятельности	Оценочное средство		
	Тестовые задания	Практические работы	Итого
Лекции	20		20
Практические занятия		80	80
Итого	20	80	100

Сумма баллов, набранных студентом по всем видам учебной деятельности в рамках дисциплины, переводится в оценку в соответствии с таблицей.

Сумма баллов по дисциплине	Оценка по промежуточной аттестации	Характеристика качества сформированности компетенции
от 91 до 100	«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций, обнаруживает всестороннее, систематическое и глубокое знание учебного материала, усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой, умеет свободно выполнять практические задания, предусмотренные программой, свободно оперирует приобретенными знаниями и умениями, применяет их в ситуациях повышенной сложности.
от 76 до 90	«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
от 61 до 75	«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций: в ходе контрольных мероприятий допускаются значительные ошибки, проявляется отсутствие отдельных знаний, умений, навыков по некоторым дисциплинарным компетенциям, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
от 41 до 60	«не зачтено» / «неудовлетворительно»	У студента не сформированы дисциплинарные компетенции, проявляется недостаточность знаний, умений, навыков.
от 0 до 40	«не зачтено» / «неудовлетворительно»	Дисциплинарные компетенции не сформированы. Проявляется полное или практически полное отсутствие знаний, умений, навыков.

5 Примерные оценочные средства

5.1 Примеры тестовых заданий

1. Given:

```
public class Sequence {
    Sequence() {
        System.out.print("x ");
    }
    public static void main(String[] args) {
        new Sequence().go();
    }
    void go() {
        System.out.print("y ");
    }
}
```

What is the result?

- A) x y
- B) y x
- C) x c
- D) y c

2. Given:

```
public class MyStuff {
    MyStuff(String n) {
        name = n;
    }
    String name;
    public static void main(String[] args) {
        MyStuff m1 = new MyStuff("guitar");
        MyStuff m2 = new MyStuff("tv");
        System.out.println(m2.equals(m1));
    }
}
```

```

    }
    public boolean equals(Object o) {
        MyStuff m = (MyStuff) o;
        if(m.name != null)
            return true;
        return false;
    }
}

```

What is the result?

- A) The output is "true" and MyStuff fulfills the Object.equals() contract.
- B) The output is "false" and MyStuff fulfills the Object.equals() contract.
- C) The output is "true" and MyStuff does NOT fulfill the Object.equals() contract.
- D) The output is "false" and MyStuff does NOT fulfill the Object.equals() contract
- E) Compilation fails

3. Given:

```

public class MyLoop {
    public static void main(String[] args) {
        String[] sa = {"tom ", "jerry "};
        for(int x = 0; x
            for(String s: sa) {
                System.out.print(x + " " + s);
                if( x == 1) break;
            }
        }
    }
}

```

What is the result?

- A) 0 tom 0 jerry 1 tom
- B) 0 tom 0 jerry 1 tom 1 jerry
- C) 0 tom 0 jerry 2 tom 2 jerry
- D) 0 tom 0 jerry 1 tom 2 tom 2 jerry
- E) 0 tom 0 jerry 1 tom 1 jerry 2 tom 2 jerry

4. Given:

```

class Feline {
    public String type = "f ";
    public Feline() {
        System.out.print("feline ");
    }
}
public class Cougar extends Feline {
    public Cougar() {
        System.out.print("cougar ");
    }
    public static void main(String[] args) {
        new Cougar().go();
    }
    void go() {
        type = "c ";
        System.out.print(this.type + super.type);
    }
}

```

What is the result?

- A) cougar c c

- B) cougar c f
- C) feline cougar c c
- D) feline cougar c f
- E) Compilation fails
- F) An exception is thrown at run time.

5. Given:

```
class Alpha {
    String getType() {
        return "alpha";
    }
}
class Beta extends Alpha {
    String getType() {
        return "beta";
    }
}
class Gamma extends Beta {
    String getType() {
        return "gamma";
    }
    public static void main(String[] args) {
        Gamma g1 = new Alpha();
        Gamma g2 = new Beta();
        System.out.println(g1.getType() + " "
            + g2.getType());
    }
}
```

What is the result?

- A) alpha beta
- B) beta beta
- C) gamma gamma
- D) alpha alpha
- E) Compilation fails.

6. Choose the False statement among the following?

- Java support globally accessible variables
- Java is programming language from Sun
- JavaScript is a programming language from Netscape
- None of the Above

7. The 16 bit coding scheme employed in Java programming is

- EDCDIC
- UNICODE
- ASCII
- Hexadecimal

8. What is the output of the following program?

```
public class example {
    public static void main(String args[]) {
        int x=0, y=2;
        do {
            x=++x;
            y--;
        } while(y>0);
        System.out.println(x);
    }
}
```

```
}  
0  
1  
2
```

Compilation Error

9. Command to execute a compiled java programs is

run

execute

javac

java

10. Choose the correct statement from the given choice?

Arrays in Java are essentially objects

It is not possible to assign one array to another. Individual elements of array can however be assigned.

Array elements are indexed from 1 to size of array

If a method tries to access an array element beyond its range, a compile warning is generated.

11. Consider the following code segment.

```
List list = new ArrayList();
```

```
list.add("P");
```

```
list.add("Q");
```

```
list.add("R");
```

```
list.set(2, "s");
```

```
list.add(2, "T");
```

```
list.add("u");
```

```
System.out.println(list);
```

What is printed as a result of executing the code segment?

(a) [P, Q, R, s, T]

(b) [P, Q, s, T, u]

(c) [P, Q, T, s, u]

(d) [P, T, Q, s, u]

(e) [P, T, s, R, u]

12. Consider the following instance variable and method.

```
private List nums;
```

```
/** Precondition: nums.size > 0
```

```
*/
```

```
public void numQuest() {
```

```
    int k = 0;
```

```
    Integer zero = new Integer(0);
```

```
    while (k
```

```
        if (nums.get(k).equals(zero))
```

```
            nums.remove(k);
```

```
            k++;
```

```
    }
```

```
}
```

Assume that List nums initially contains the following

Integer values.

[0, 0, 4, 2, 5, 0, 3, 0]

What will List nums contain as a result of executing numQuest ?

(a) [0, 0, 4, 2, 5, 0, 3, 0]

(b) [4, 2, 5, 3]

(c) [0, 0, 0, 0, 4, 2, 5, 3]

(d) [3, 5, 2, 4, 0, 0, 0, 0]

(e) [0, 4, 2, 5, 3]

13. At a certain high school students receive letter grades based on the following scale.

Numeric Score Letter Grade

93 or above A

From 84 to 92 inclusive B

From 75 to 83 inclusive C

Below 75 F

Which of the following code segments will assign the correct string to grade for a given integer score ?

I.

```
if (score => 93)
    grade 5 "A";
if (score >= 84 && score
    grade 5 "B";
if (score >= 75 && score
    grade 5 "C";
if (score , 75)
    grade
```

II.

```
if (score >= 93)
    grade 5 "A";
if (84
    grade 5 "B";
if (75
    grade 5 "C";
if (score
    grade 5 "F";
```

III.

```
if (score >= 93)
    grade 5 "A";
else if (score >= 84)
    grade 5 "B";
else if (score >= 75)
    grade 5 "C";
else
    grade 5 "F";
```

(a) II only

(b) III only

(c) I and II only

(d) I and III only

(e) I, II, and III

14. Consider the following declarations.

```
public interface Comparable {
    int compareTo(Object other);
}
public class SomeClass implements Comparable {
    // ... other methods not shown
}
```

Which of the following method signatures of compareTo will satisfy the Comparable interface requirement?

I. public int compareTo(Object other)

II. public int compareTo(SomeClass other)

III. public boolean compareTo(Object other)

- (a) I only
- (b) II only
- (c) III only
- (d) I and II only
- (e) I, II, and III

15. Consider the following instance variable and method.

```
private int[] myStuff;  
/** Precondition: myStuff contains int values in no particular order.  
*/  
public int mystery(int num) {  
    for (int k = myStuff.length - 1; k >= 0; k--) {  
        if (myStuff[k]  
            return k;  
        }  
    }  
    return -1;  
}
```

Which of the following best describes the contents of myStuff after the following statement has been executed?

```
int m = mystery(n);
```

- (a) All values in positions 0 through m are less than n.
- (b) All values in positions m+1 through myStuff.length-1 are less than n.
- (c) All values in positions m+1 through myStuff.length-1 are greater than or equal to n.
- (d) The smallest value is at position m.
- (e) The largest value that is smaller than n is at position m.

16. Consider the following method.

```
/** Precondition: x >= 0  
*/  
public void mystery(int x) {  
    System.out.print(x % 10);  
    if ((x / 10) != 0) {  
        mystery(x / 10);  
    }  
    System.out.print(x % 10);  
}
```

Which of the following is printed as a result of the call mystery(1234) ?

- (a) 1441
- (b) 3443
- (c) 12344321
- (d) 43211234
- (e) Many digits are printed due to infinite recursion.

17. Consider the following two classes.

```
public class Dog {  
    public void act() {  
        System.out.print("run");  
        eat();  
    }  
    public void eat() {  
        System.out.print("eat");  
    }  
}  
  
public class UnderDog extends Dog {
```

```

public void act() {
    super.act();
    System.out.print("sleep");
}
public void eat() {
    super.eat();
    System.out.print("bark");
}
}

```

Assume that the following declaration appears in a client program.

```
Dog fido = new UnderDog();
```

What is printed as a result of the call `fido.act()` ?

- (a) run eat
- (b) run eat sleep
- (c) run eat sleep bark
- (d) run eat bark sleep
- (e) Nothing is printed due to infinite recursion.

18. Consider the following recursive method.

```

public static int mystery(int n) {
    if (n == 0)
        return 1;
    else
        return 3 * mystery(n - 1);
}

```

What value is returned as a result of the call `mystery(5)` ?

- (a) 0
- (b) 3
- (c) 81
- (d) 243
- (e) 6561

19. Consider the following instance variable and method.

```

private int[] arr;
/** Precondition: arr.length > 0
 */
public int checkArray() {
    int loc = arr.length / 2;
    for (int k = 0; k < arr.length; k++)
        if (arr[k] > arr[loc])
            loc = k;
    return loc;
}

```

Which of the following is the best postcondition for `checkArray` ?

- (a) Returns the index of the first element in array `arr` whose value is greater than `arr[loc]`
- (b) Returns the index of the last element in array `arr` whose value is greater than `arr[loc]`
- (c) Returns the largest value in array `arr`
- (d) Returns the index of the largest value in array `arr`
- (e) Returns the index of the largest value in the second half of array `arr`

20. Assume the following declarations have been made.

```

private String s;
private int n;
public void changer(String x, int y)
{

```

```

x = x + "peace";
y = y * 2;
}

```

Assume s has the value "world" and n is 6. What are the values of s and n after the call changer(s, n) ?

- (a) a is world, n is 6
- (b) a is worldpeace, n is 6
- (c) a is world, n is 12
- (d) a is worldpeace, n is 12
- (e) a is peace n is 12

Краткие методические указания

Промежуточный тест проводится в электронной форме во время последнего в учебном периоде практического занятия. Тест состоит из 20 тестовых заданий. На выполнение теста отводится 20 минут. Во время проведения теста использование литературы и других информационных ресурсов допускается только по предварительному согласованию с преподавателем.

Шкала оценки

Оценка	Баллы	Описание
5	19–20	Процент правильных ответов от 95% до 100%
4	16–18	Процент правильных ответов от 80 до 94%
3	13–15	Процент правильных ответов от 65 до 79%
2	9–12	Процент правильных ответов от 45 до 64%

5.2 Примеры заданий для выполнения практических работ

Тема 1. Синтаксис языка Java, классы в языке Java.

Тема 2. Наследование и инкапсуляция в языке Java.

Тема 3. Разработка классов в языке Java.

Тема 4. Наследование и интерфейсы в языке Java.

Тема 5. Обобщённые типы и коллекции значений в языке Java.

Тема 6. Работа со строками.

Тема 7. Обработка исключений.

Тема 8. Поток ввода/вывода.

Тема 9. Многопоточные программы Java.

Тема 10. Параллельное программирование Java.

Тема 11. Построение приложений баз данных с использованием JDBC API.

Тема 12. Локализация Java программ.

Краткие методические указания

На выполнение одной практической работы отводится не более одного двухчасового занятия (не включая затраты времени на проведение промежуточного теста на последнем в учебном периоде практическом занятии). После выполнения каждой практической работы студент должен представить отчет о ее выполнении, а также, по указаниям преподавателя, выполнить дополнительные практические задания по теме работы.

Шкала оценки

Оценка	Баллы	Описание
5	73–80	Студент демонстрирует умения на итоговом уровне: умеет свободно выполнять практически все задания, предусмотренные программой, свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.
4	61–72	Студент демонстрирует умения на среднем уровне: освоил основные умения, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе умений на новые, нестандартные ситуации.
3	49–60	Студент демонстрирует умения и навыки на базовом уровне: в ходе контрольных мероприятий допускаются значительные ошибки, проявляется отсутствие отдельных умений, навыков по дисциплинарной компетенции, испытываются значительные затруднения при оперировании умениями и при их переносе на новые ситуации.

2	33–48	Студент демонстрирует умения и навыки на уровне ниже базового: проявляется недостаточность умений и навыков.
---	-------	--