МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Рязанский государственный радиотехнический университет»

Кафедра «Безопасность жизнедеятельности и экология»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине

«Параллельное программирование»

Направление

09.03.04 Программная инженерия

Профиль

Программная инженерия систем искусственного интеллекта
Программное обеспечение систем искусственного интеллекта

Квалификация выпускника – бакалавр

Форма обучения – очная

Рязань

1. ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов и процедур, предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретённых компетенций обучающихся целям и требованиям основной образовательной программы в ходе текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности компетенций и индикаторов их достижения, приобретаемых обучающимися в соответствии с требованиями ФГОС ВО.

Контроль знаний обучающихся проводится в форме текущего контроля (лабораторные работы) и промежуточной аттестации (экзамен).

2. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ С УКАЗАНИЕМ ЭТАПОВ ИХ ФОРМИРОВАНИЯ

При освоении дисциплины формируются следующие компетенции:

- **ПК-10** – способен применять различные модели и (или) технологии обработки данных (BD-4);
  - **ПК-10.1** – осуществляет выбор технологий обработки больших данных, приемлемых для создания прикладной системы ИИ с заданными требованиями (BD-4.1);
  - **ПК-10.2** – разрабатывает и отлаживает прикладные решения с элементами ИИ с применением различных технологий обработки данных (BD-4.2);
  - **ПК-10.3** – тестирует, испытывает и оценивает качество решений с элементами ИИ, реализованных с использованием технологий обработки данных (BD-4.3).
- **ПК-16** – способен применять JVM-совместимые языки программирования для решения задач в области ИИ (PL-2);
  - **ПК-16.1** – разрабатывает и отлаживает прикладные решения разного уровня сложности с использованием JVM-совместимых языков программирования, тестирует, испытывает и оценивает качество таких решений (PL-2.1).

Указанные компетенции формируются в соответствии со следующими этапами:

- формирование и развитие теоретических знаний (лекционные занятия, самостоятельная работа);
- приобретение и развитие практических умений (лабораторные работы, самостоятельная работа);

- закрепление знаний и навыков (выполнение и защита лабораторных работ, подготовка к экзамену).

3. ПОКАЗАТЕЛИ И КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформированность каждой компетенции оценивается по трёхуровневой шкале:

- **пороговый уровень** – обязательный для всех обучающихся; соответствует освоению более 40% приведённых знаний, умений и навыков;
- **продвинутый уровень** – соответствует освоению более 60% знаний, умений и навыков;
- **эталонный уровень** – соответствует освоению более 80% знаний, умений и навыков.

При освоении менее 40% компетенция считается несформированной.

Таблица 1. Показатели достижения индикаторов компетенций

Компетенция	Индикаторы (краткое содержание)	Этап	Оценочное средство
ПК-10.1	Знать современные технологии параллельной обработки больших данных; уметь выполнять параллельную обработку больших данных; владеть инструментарием параллельной обработки больших данных.	6	Экзамен
ПК-10.2	Знать технологии параллельной обработки данных; уметь разрабатывать и отлаживать прикладные решения с элементами ИИ с применением технологий параллельной обработки данных; владеть инструментальными средствами организации параллельной обработки данных.	6	Экзамен
ПК-10.3	Знать технологии тестирования и отладки решений с параллельной и потоковой обработкой данных; уметь тестировать и отлаживать решения; владеть инструментами тестирования и отладки.	6	Экзамен

Компетенция	Индикаторы (краткое содержание)	Этап	Оценочное средство
ПК-16.1	Знать модель памяти Java, методы разработки, анализа и оптимизации параллельных и распределённых программных систем; уметь поддерживать приложения с высоким параллелизмом и конкуренцией; владеть навыками применения простейших примитивов многопоточного программирования.	6	Экзамен

4. ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Наименование оценочного средст
Тема 1. Введение в параллельное программирование. Параллельные вычислительные системы	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 2. Основы многопоточности в Java. Модель памяти	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 3. Синхронизация потоков. Примитивы синхронизации	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 4. Продвинутые механизмы синхронизации в Java (пакеты, атомарные классы, коллекции)	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 5. Высокоуровневые абстракции параллелизма (Executor, Fork/Join, параллельные потоки)	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 6. Введение в потоковую обработку. Apache Kafka, Apache Flink	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Наименование оценочного средст
Тема 7. Обработка больших данных: архитектура Hadoop и MapReduce	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Тема 8. Паттерны параллельного программирования. Тестирование многопоточных приложений	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Экзамен
Лабораторный практикум (8 лабораторных работ)	ПК-10.1, ПК-10.2, ПК-10.3, ПК-16.1	Защита лабораторн работ

5. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ

5.1. Промежуточная аттестация (экзамен)

Экзамен проводится в форме устного ответа по билетам, включающим теоретический вопрос и практическое задание (задачу). Также может использоваться тестирование.

5.1.1. Тестовые вопросы закрытого типа (20 вопросов)

1. Что такое параллельное программирование?
а) Программирование, при котором несколько вычислений выполняются одновременно;
б) Программирование в одном потоке;
в) Программирование баз данных;
г) Программирование пользовательских интерфейсов.
Правильный ответ: а)

2. Какая модель памяти используется в Java?
а) Модель памяти Java (JMM);
б) Модель памяти C++;
в) Модель памяти Python;
г) Модель памяти SQL.
Правильный ответ: а)

3. Что такое поток (thread) в Java?
а) Наименьшая единица выполнения, которая может выполняться параллельно с другими потоками;

б) Процесс;
в) Класс;
г) Пакет.
Правильный ответ: а)

4. Какой примитив синхронизации в Java используется для блокировки?
а) synchronized;
б) volatile;
в) transient;
г) static.
Правильный ответ: а)

5. Что такое deadlock (взаимная блокировка)?
а) Ситуация, когда два или более потока бесконечно ожидают друг друга;
б) Ошибка компиляции;
в) Тип исключения;
г) Метод оптимизации.
Правильный ответ: а)

6. Что такое race condition (состояние гонки)?
а) Ситуация, когда результат зависит от порядка выполнения потоков;
б) Ошибка ввода-вывода;
в) Тип коллекции;
г) Метод синхронизации.
Правильный ответ: а)

7. Какой класс из пакета java.util.concurrent используется для создания пула потоков?
а) ExecutorService;
б) Thread;
в) Runnable;
г) Object.
Правильный ответ: а)

8. Что такое Fork/Join Framework?
а) Фреймворк для параллельного выполнения рекурсивных задач;
б) Фреймворк для работы с базами данных;
в) Фреймворк для веб-разработки;
г) Фреймворк для тестирования.
Правильный ответ: а)

9. Что такое атомарные классы в Java?
а) Классы, операции над которыми выполняются атомарно без блокировок;
б) Классы для работы с файлами;
в) Классы для сетевого взаимодействия;
г) Классы для шифрования.
Правильный ответ: а)

10. Что такое Apache Kafka?
а) Распределённый брокер сообщений для потоковой обработки данных;
б) База данных;
в) Веб-сервер;
г) Фреймворк для машинного обучения.
Правильный ответ: а)

11. Что такое Apache Flink?
а) Фреймворк для потоковой обработки данных с поддержкой параллелизма;
б) Реляционная СУБД;
в) Операционная система;
г) Язык программирования.
Правильный ответ: а)

12. Что такое MapReduce?
а) Модель параллельных вычислений для обработки больших данных;
б) Тип базы данных;
в) Протокол передачи данных;
г) Метод шифрования.
Правильный ответ: а)

13. Что такое Hadoop?
а) Фреймворк для распределённой обработки больших данных;
б) Веб-браузер;
в) Текстовый редактор;
г) Антивирус.
Правильный ответ: а)

14. Что такое Producer-Consumer (производитель-потребитель)?
а) Паттерн параллельного программирования, при котором одни потоки производят данные, другие потребляют;
б) Тип базы данных;
в) Метод сортировки;
г) Алгоритм шифрования.
Правильный ответ: а)

15. Что такое thread-safe (потокобезопасность)?
а) Свойство кода, гарантирующее корректную работу при многопоточном доступе;
б) Свойство кода, работающего только в одном потоке;
в) Тип исключения;
г) Метод оптимизации.
Правильный ответ: а)

16. Какой метод используется для запуска потока в Java?
а) start();
б) run();
в) execute();

г) launch().
Правильный ответ: а)

17. Что такое volatile в Java?
а) Ключевое слово, гарантирующее видимость изменений переменной между потоками;
б) Тип данных;
в) Метод синхронизации;
г) Класс коллекции.
Правильный ответ: а)

18. Что такое ThreadLocal?
а) Класс, предоставляющий локальную переменную для каждого потока;
б) Тип пула потоков;
в) Метод синхронизации;
г) Класс для работы с файлами.
Правильный ответ: а)

19. Что такое CompletableFuture?
а) Класс для асинхронного программирования в Java;
б) Тип коллекции;
в) Метод синхронизации;
г) Класс для работы с сетью.
Правильный ответ: а)

20. Что такое параллельный поток (parallel stream) в Java?
а) Поток данных, обрабатываемый параллельно с использованием нескольких ядер;
б) Обычный поток ввода-вывода;
в) Тип коллекции;
г) Метод сортировки.
Правильный ответ: а)

5.1.2. Тестовые вопросы открытого типа (10 вопросов)

1. Опишите модель памяти Java (JMM) и её основные принципы (видимость, атомарность, упорядоченность).

2. Объясните разницу между процессами и потоками. Приведите примеры использования.

3. Опишите основные примитивы синхронизации в Java (synchronized, Lock, Semaphore, CountDownLatch).

4. Что такое deadlock? Опишите условия возникновения и способы предотвращения.

5. Опишите паттерн Producer-Consumer и его реализацию в Java.

6. Что такое Fork/Join Framework? Приведите пример его использования.

7. Опишите архитектуру Apache Kafka и её основные компоненты (Producer, Consumer, Broker, Topic, Partition).

8. Сравните Apache Kafka и Apache Flink по назначению и областям применения.

9. Опишите модель MapReduce и её применение для обработки больших данных.

10. Какие методы тестирования и профилирования многопоточных приложений вы знаете? Приведите примеры инструментов.

5.1.3. Типовые практические задания (задачи) к экзамену

1. Разработать потокобезопасный счётчик на Java с использованием synchronized.
2. Реализовать модель «Обедающие философы» с предотвращением deadlock.
3. Реализовать thread-safe кэш с использованием ConcurrentHashMap.
4. Реализовать асинхронную обработку задач с использованием ExecutorService.
5. Реализовать паттерн Producer-Consumer с использованием BlockingQueue.
6. Интегрировать приложение с Apache Kafka (Producer и Consumer).
7. Реализовать обработку потока событий в Apache Flink.
8. Провести тестирование многопоточного приложения под нагрузкой.
9. Реализовать параллельную обработку данных с использованием Fork/Join Framework.
10. Реализовать параллельную сортировку массива с использованием параллельных потоков Java.

5.1.4. Вопросы к экзамену (30 вопросов)

1. Введение в параллельное программирование. Параллельные вычислительные системы.
2. Общая характеристика многопроцессорных систем.
3. Основы многопоточности в Java. Модель памяти Java (JMM).
4. Создание и запуск потоков в Java. Класс Thread и интерфейс Runnable.
5. Синхронизация потоков. Примитивы синхронизации (synchronized, volatile).
6. Продвинутые механизмы синхронизации: Lock, ReentrantLock, ReadWriteLock.
7. Атомарные классы (AtomicInteger, AtomicReference и др.).
8. Коллекции для многопоточности (ConcurrentHashMap, CopyOnWriteArrayList, BlockingQueue).
9. Высокоуровневые абстракции параллелизма: Executor Framework.
10. Fork/Join Framework. Параллельные потоки (parallel streams).
11. Настройка пула потоков. ThreadPoolExecutor.
12. Паттерны параллельного программирования: Producer-Consumer, Thread Pool, Future.
13. Deadlock, livelock, starvation. Способы предотвращения.
14. Введение в потоковую обработку данных.
15. Apache Kafka: архитектура, компоненты, применение.
16. Apache Flink: архитектура, компоненты, применение.
17. Обработка больших данных: архитектура Hadoop.
18. Модель MapReduce. Применение для обработки больших данных.

19. Тестирование многопоточных приложений. Инструменты и подходы.
20. Профилирование и отладка многопоточных приложений.
21. Асинхронное программирование в Java (CompletableFuture).
22. Реактивное программирование. Основы.
23. Событийно-ориентированный параллелизм.
24. Проблемы производительности в многопоточных приложениях.
25. Оптимизация параллельных вычислений.
26. Многопоточность в базах данных.
27. Распределённые вычисления. Основы.
28. Параллельная обработка данных в системах ИИ.
29. Инструменты мониторинга и управления потоками.
30. Перспективные направления развития параллельного программирования.

6. КРИТЕРИИ ОЦЕНИВАНИЯ

6.1. Шкала оценивания тестирования

Шкала	Критерий
5 (отлично)	Процент верных ответов от 90 до 100%
4 (хорошо)	Процент верных ответов от 75 до 89%
3 (удовлетворительно)	Процент верных ответов от 60 до 74%
2 (неудовлетворительно)	Процент верных ответов менее 60%

6.2. Шкала оценивания теоретического вопроса на экзамене

Шкала	Критерий
5 (отлично)	Дан полный ответ, показаны глубокие систематизированные знания, приведены примеры, даны ответы на дополнительные вопросы.
4 (хорошо)	Дан полный ответ, но на некоторые дополнительные вопросы ответы даны с помощью наводящих вопросов.
3 (удовлетворительно)	Дан неполный ответ, на дополнительные вопросы ответы даны только с помощью преподавателя.

Шкала	Критерий
2 (неудовлетворительно)	Студент не смог ответить на вопрос.

6.3. Шкала оценивания практического задания на экзамене

Шкала	Критерий
5 (отлично)	Задание решено верно, получены корректные результаты, сделаны выводы.
4 (хорошо)	Задание решено верно, но имеются незначительные технические неточности.
3 (удовлетворительно)	Задание решено верно с дополнительными наводящими вопросами преподавателя.
2 (неудовлетворительно)	Задание не решено.

6.4. Итоговая оценка на экзамене

На экзамене студент отвечает на теоретический вопрос и выполняет практическое задание. Максимальная сумма баллов – 10 (5 за теорию + 5 за практику).
Итоговая оценка выставляется по пятибалльной шкале:

- «отлично» – 9–10 баллов;
- «хорошо» – 7–8 баллов;
- «удовлетворительно» – 5–6 баллов;
- «неудовлетворительно» – менее 5 баллов.

Оператор ЭДО ООО "Компания "Тензор"
ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
СОГЛАСОВАНО **ФГБОУ ВО "РГРТУ", РГРТУ,** Овечкин Геннадий Владимирович, Заведующий кафедрой ВПМ **05.11.26** 14:46 (MSK) Простая подпись