

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ ВЛАДИМИРА ДАЛЯ»

Факультет компьютерных систем и информационных технологий

Кафедра информационных и управляющих систем

УТВЕРЖДАЮ

Декан факультета компьютерных  
систем и информационных  
технологий



Кочевский А.А.

« 19 » \_\_\_\_\_ 2023 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
по учебной дисциплине

«Построение распределённых систем мониторинга»

09.03.02 Информационные системы и технологии

«Информационные системы и технологии»

Разработчик:

доцент \_\_\_\_\_ Киреев И.Ю.

ФОС рассмотрен и одобрен на заседании кафедры информационных и управляющих систем от «18» апреля 2023 г., протокол № 15

Заведующий кафедрой

информационных и управляющих систем \_\_\_\_\_ Горбунов А.И.

Луганск 2023 г.

**1. Паспорт**  
**фонда оценочных средств по учебной дисциплине**  
**«Построение распределенных систем мониторинга»**

---

Перечень компетенций (элементов компетенций), формируемых в результате освоения учебной дисциплины (модуля) или практики

| № п/п | Код контролируемой компетенции | Формулировка контролируемой компетенции                                                                                      | Контролируемые разделы (темы) учебной дисциплины                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Этапы формирования (семестр изучения) |
|-------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------|
| 1     | 2                              | 3                                                                                                                            | 4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 5                                     |
| 1     | ОПК-7                          | Знать: основные платформы, технологии и инструментальные программно-аппаратные средства для реализации информационных систем | Тема 1. Введение в архитектуру распределенных информационных систем.<br>Тема 2. Введение в параллельные вычисления.<br>Тема 3. Обзор параллельных вычислительных систем и их классификация.<br>Способы организации параллельной обработки данных.<br>Тема 4. Вычислительные кластеры: основные понятия, архитектура, типовый набор кластерного программного обеспечения, средства доступа и управления, тестирование производительности.<br>Тема 5. Моделирование параллельных программ. Реализация параллелизма различного вида. Общая схема и методика разработки параллельных алгоритмов.<br>Тема 6. Базовые средства параллельного программирования вычислительных кластеров. Методы | 7                                     |

|  |  |  |                                                                                                                                                                                                                                                                                                                                                                              |  |
|--|--|--|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
|  |  |  | <p>передачи данных.<br/>Стандарт MPI.<br/>Тема 7. Базовые средства параллельного программирования систем с общей памятью.<br/>Стандарт OpenMP.<br/>Тема 8 8.<br/>Высокоуровневые средства программирования многопроцессорных систем. DVM-система.<br/>Тема 9.<br/>Высокопроизводительные вычисления с применением графических процессоров (GPU). Технология NVidia CUDA.</p> |  |
|--|--|--|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|

*Примечание:*

\* Наименование темы (раздела) или тем (разделов) берется из рабочей программы учебной дисциплины (программы практики) в соответствии с компетенциями, которые формируются у студента при ее изучении.

## 2. Показатели и критерии оценивания компетенций, описание шкал оценивания

Таблица 2.1 – Показатели оценивания компетенций

| № п/п | Код контролируемой компетенции | Показатель оценивания (знания, умения, навыки)                                                                                                                                                                                                | Контролируемые разделы (темы) учебной дисциплины (модуля), практики <sup>1</sup>                                                                                                                                                                                     | Наименование оценочного средства <sup>2</sup> |
|-------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| 1     | 2                              | 3                                                                                                                                                                                                                                             | 4                                                                                                                                                                                                                                                                    | 5                                             |
| 1     | ОПК-7                          | <p>Знать: основные платформы, технологии и инструментальные программно-аппаратные средства для реализации информационных систем</p> <p>Уметь: осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации</p> | <p>Тема 1. Введение в архитектуру распределенных информационных систем.</p> <p>Тема 2. Введение в параллельные</p> <p>Тема 3. Обзор параллельных вычислительных систем и их классификация.</p> <p>Способы организации параллельной обработки данных. вычисления.</p> | Лабораторная работа                           |

|  |  |                                                                                                                                                                                                                             |  |  |
|--|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|--|
|  |  | информационных систем, применять современные технологии реализации информационных систем.<br>Иметь навыки: владения технологиями и инструментальными программно-аппаратными средствами для реализации информационных систем |  |  |
|--|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|--|

*Примечание:*

<sup>1</sup> - Контролируемые разделы (темы) учебной дисциплины, перечисленные в рабочей программе, показывающие этапность формирования компетенции в ходе изучения учебной дисциплины.

<sup>2</sup> - Наименование оценочного средства в соответствии с их примерным перечнем.

### **Вопросы для защиты лабораторных работ**

Лабораторная работа №1. Методы создания потоков в Java.

*Цель работы:* приобрести навыки разработки многопоточных программ в Java.

Контрольные вопросы:

1. Какие методы создания потоков в Java Вы знаете? Опишите каждый метод.
2. С какой целью используется метод run() класса Thread? Укажите его сигнатуру.
3. Как запустить поток на выполнение? Какой метод для этого используется?
4. Укажите условия завершения многопоточной Java программы

Лабораторная **работа №2. Получение и установка параметров потоков.**

*Цель работы:* изучить состав параметров потоков и методы работы с ними.

Контрольные вопросы:

1. Перечислите информационные поля потока.
2. Укажите диапазон допустимых значений приоритета потока (класса Thread).
3. Какое значение соответствует самому низкому приоритету?
4. Какое значение соответствует самому высокому приоритету?
5. В каких состояниях может находиться поток?

Лабораторная работа №3. Исследование методов прерывания потоков.

*Цель работы:* приобрести навыки разработки программ прерывания потоков.

Контрольные вопросы:

1. Назовите условия завершения работы многопоточной Java программы
2. Что такое поток – демон?

3. Каким образом поток может отреагировать на прерывание с требованием завершения его работы?
4. Какие методы потока используются для проверки факта его прерывания?
5. Что общего и в чем различие методов **isInterrupted()** и **interrupted()**?
6. Может ли поток проигнорировать требование своего завершения? Какая реакция потока на запрос завершения работы является рекомендуемой?

Лабораторная работа №4. Работа с исключениями в потоках

*Цель работы:* Приобрести навыки обработки прерываний в потоках в Java приложениях.

Контрольные вопросы:

1. Какое исключение предусматривается в Java для использования в потоках?
2. Для какой цели используется исключение **InterruptedException**?
3. В каком методе потока рекомендуется обрабатывать исключение **InterruptedException**?
4. Назовите методы библиотеки Java, которые могут бросать исключение **InterruptedException**.
5. Объясните механизм работы рекурсии, который использован в лабораторной работе.

Лабораторная работа №5. Приостановка работы потока

*Цель работы:* работы: приобрести навыки использования в Java программного механизма приостановки работы потока.

Контрольные вопросы:

1. Для чего используется класс **TimeUnit**?
2. Какие у него есть атрибуты для указания отрезков времени?
3. Для чего используется метод **sleep()**?
4. Что является рекомендуемой практикой программирования при обработке прерываний остановки потока?
5. Как связан вызов метода **sleep()** с использованием потоком процессорного времени?
6. Какое исключение может бросить метод **sleep()**? Назовите условия бросания этого исключения. Исключение бросается сразу при возникновении необходимых условий или поток ожидает своего выхода из «состояния сна»?
7. Для каких целей используется метод **yield()**?

Лабораторная работа №6. Ожидание завершения работы потока

*Цель работы:* приобрести навыки разработки многопоточного программного обеспечения с использованием аппарата ожидания завершения работы потока.

Контрольные вопросы:

1. Какому классу принадлежит метод **join()**?
2. Опишите назначение метода **join()**.
3. Какие варианты метода **join()** Вы знаете?
4. Опишите структуру и алгоритм работы программы, разработанной в рамках данной лабораторной работы

Лабораторная работа №7. Создание и запуск потоков-демонов

*Цель работы:* приобрести навыки разработки многопоточного программного обеспечения с использованием потоков-демонов.

Контрольные вопросы:

1. Для решения каких задач используются потоки- демоны?
2. Назовите характеристики (особенности) потока-демона
3. Когда завершаются потоки демоны? (Назовите условия завершения).

4. Как создается поток-демон?
5. Для чего можно использовать метод **isDaemon()**?
6. Какой приоритет имеют потоки-демоны?

Лабораторная работа №8. Обработка исключительных ситуаций в потоках

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием обработчиков исключительных ситуаций.

Контрольные вопросы:

1. Дайте характеристику классификации исключений в Java.
2. Укажите способы обработки **Checked** исключений в программе?
3. Укажите способы обработки **Unchecked** исключений в программе?
4. Дайте описание способов обработки прерываний в потоках Java.
5. Укажите алгоритм обработки непроверяемых исключений в потоках Java.

Лабораторная работа №9. Использование локальных переменных потока

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием локальных переменных потока.

Контрольные вопросы:

1. Опишите проблему разделения полей в потоках. Для каких потоков эта проблема актуальна?
2. Укажите способ преодоления проблемы разделения переменных в потоках?
3. Для какой цели используется механизм **thread-local**? Приведите примеры его использования.

Лабораторная работа №10. Создание групп потоков

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием групп потоков.

Контрольные вопросы:

1. Для какой цели используется класс **ThreadGroup** ?
2. Как указать, что поток относится к данной группе потоков?
3. Перечислите методы класса **ThreadGroup**. Для этого проконсультируйтесь в документации на классы Java.
4. Как в рамках лабораторной работы решена задача принудительного завершения работы потоков, входящих в группу?
5. Для какой цели в программе используется объект класса **Random**?
6. Каким образом в программе обнаруживается факт завершения работы одного (первого) потока?

Лабораторная работа №11. Обработка непроверяемых исключений в группе потоков

*Цель работы:* приобрести навыки разработки многопоточных приложений с поддержкой возможности обработки непроверяемых исключений в группе потоков.

Контрольные вопросы:

1. Какие виды исключений предусмотрены в Java. Укажите отличия **Checked** и **Unchecked** исключений.
2. Опишите особенности обработки исключений в группах потоков **ThreadGroup**.
3. Охарактеризуйте особенности работы программы, созданной в рамках данной лабораторной работы.

Лабораторная работа №12. Создание потоков при помощи фабрики

*Цель работы:* приобрести навыки разработки многопоточных приложений, в которых потоки создаются на основе паттерна «фабрика».

Контрольные вопросы:

1. Опишите особенности паттерна «фабрика».
2. Как паттерн «фабрика» используется для создания объектов- потоков?
3. Какие параметры получает метод `newThread`? Какой результат он возвращает?
4. Перечислите преимущества использования паттерна «фабрика» для создания объектов – потоков.

Лабораторная работа №13. Синхронизация методов и участков кода

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием синхронизированных методов.

Контрольные вопросы:

1. Опишите проблемы доступа к общим данным в многопоточном приложении.
2. Что такое «критический участок» и как этот механизм реализован в Java?
3. Что такое синхронизированный метод? Как такой метод описывается в Java? Опишите порядок работы конкурирующих потоков в синхронизированном методе?
4. Укажите особенности статических синхронизированных методов.
5. Если один поток захватил один синхронизированный метод объекта, может ли другой поток захватить другой синхронизированный метод этого же объекта?
6. Объясните работу программы, разработанной в рамках данной лабораторной работы, с использованием синхронизированных методов и без использования синхронизированных методов.
7. Что такое синхронизированный участок кода? Какой параметр нужно передавать синхронизированному участку кода?

Лабораторная работа №14. Обработка независимых атрибутов в синхронизированных классах

*Цель работы:* приобрести навыки разработки многопоточных приложений с реализацией безопасного одновременного доступа к независимым атрибутам объектов.

Контрольные вопросы:

1. Опишите метод ограничения доступа в многопоточном приложении к участку кода программы.
2. Опишите алгоритм работы программы, созданной в рамках данной лабораторной работы.
3. Каким образом осуществляется доступ к двум независимым полям объекта в программе?
4. С какой целью в синхронизированный блок кода передается параметр – объект?

Лабораторная работа №15. Использование условий в синхронизированных критических участках многопоточного приложения.

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием условий в синхронизированных критических участках.

Контрольные вопросы:

1. Опишите назначение методов **`wait()`**, **`notify()`**, **`notifyAll()`**.
2. Опишите особенности использования методов **`wait()`**, **`notify()`**, **`notifyAll()`**. Как эти методы соотносятся с ключевым словом `synchronize`.
3. Опишите задачу производитель - потребитель
4. Дайте характеристику алгоритма работы программы, созданной в рамках лабораторной работы.
5. Опишите алгоритм работы методов **`get()`** и **`set()`**.

Лабораторная работа №16. Синхронизация выполнения блока программы с помощью механизма защепок (Lock).

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием синхронизированных критических участков, построенных на основе защепок.

Контрольные вопросы:

1. Опишите преимущества использования защепок для поддержания синхронизированного доступа к критическим участкам кода.
2. Укажите разницу в работе методов **tryLock()** и **lock()**.
3. Дайте характеристику алгоритма работы программы, созданной в рамках лабораторной работы.
4. Как производится освобождение защелки потоком?
5. Что такое тупиковая ситуация? Приведите пример возникновения тупиковой ситуации.

Лабораторная работа №17. Синхронизация доступа к данным в многопоточных приложениях с помощью механизма защепок чтения / записи.

*Цель работы:* приобрести навыки разработки многопоточных приложений с использованием синхронизированных критических участков, построенных на основе защепок чтения / записи.

Контрольные вопросы:

1. Опишите особенности защепок для выполнения операций чтения и записи.
2. Укажите разницу в интерфейсах Lock и ReadWriteLock.
3. Дайте характеристику алгоритма работы программы, созданной в рамках лабораторной работы.
4. Какой класс имплементирует интерфейс ReadWriteLock? Какие методы реализованы в этом классе?

Таблица 2.2 – Критерии и шкала оценивания по оценочному средству «Лабораторная работа»

| Шкала оценивания<br>(интервал баллов) <sup>2</sup> | Критерий оценивания                                                                                                                                                                                                      |
|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5                                                  | Лабораторная работа выполнена самостоятельно на высоком уровне и в полном объеме, отчет оформлен в соответствии с требованиями, сделаны правильные выводы по проведенным экспериментам.                                  |
| 4                                                  | Лабораторная работа выполнена самостоятельно на среднем уровне и в полном объеме, отчет оформлен с незначительными отклонениями от требований, допущены незначительные неточности в выводах по проведенным экспериментам |
| 3                                                  | Лабораторная работа выполнена на низком уровне и не полностью, отчет оформлен с отклонениями от требований, выводы по экспериментам сделаны не в полном объеме.                                                          |
| 2                                                  | Лабораторная работа не выполнена, отчет не оформлен, или представленный отчет не соответствует варианту задания.                                                                                                         |

Пример тем лабораторных работ.

1. Введение в параллельное программирование .Visual Studio 2010
2. Параллельное программирование в .NET Framework 4.0
3. Параллелизм доступа к данным.

4. Потокобезопасные коллекции
5. Создание Windows приложения с использованием элемента BackgroundWorker
6. Язык интегрированных запросов Parallel LINQ (PLINQ)
7. Параллельные вычисления на языке Си

### Оценочные средства для итоговой аттестации (зачет)

Перечень вопросов и заданий для зачета.

1. Теоретические вопросы.
  1. Понятия процесса и потока
  2. Класс Thread и интерфейс Runnable
  3. Взаимодействие потоков
  4. Особенности использования методов wait и notify
  5. Синхронизация потоков
  6. Поток-демон
  7. Приостановка потока
  8. Использование метода join
  9. Взаимная блокировка потоков
  10. Методы interrupt, interrupted, isInterrupted
  11. «Гонка» в многопоточном приложении
  12. «Голодание» в многопоточном приложении
  2. Пакет синхронизации java.util.concurrent
  13. Объекты синхронизации пакета java.util.concurrent
  14. Объект синхронизации Semaphore
  15. Объект синхронизации CountdownLatch
  16. Объект синхронизации CyclicBarrier
  17. Объект синхронизации Exchanger
  18. Объект синхронизации Phaser
  19. Потокобезопасные коллекции
  20. Атомарные классы пакета util.concurrent
  21. Блокирующие и неблокирующие очереди
  22. Блокировки пакета concurrent
  23. Какой метод переводит поток в режим ожидания?
  24. Какова функциональность методов notify и notifyAll?
  25. Что позволяет сделать метод join?
  26. Каковы условия вызова метода wait/notify?
  27. Дайте определение понятию “взаимная блокировка”.
  28. Чем отличаются методы interrupt, interrupted, isInterrupted?
  29. В каком случае будет выброшено исключение InterruptedException, какие методы могут его выбросить?
  30. Модификаторы volatile и метод yield().
  31. Пакет java.util.concurrent
  32. Есть некоторый метод, который выполняет операцию i++. Переменная i типа int. Предполагается, что код будет исполняться в многопоточной среде. Следует ли синхронизировать блок?
  33. Что используется в качестве mutex, если метод объявлен static synchronized? Можно ли создавать новые экземпляры класса, пока выполняется static synchronized метод?
  34. Предположим в методе run возник RuntimeException, который не был пойман. Что случится с потоком? Есть ли способ узнать о том, что Exception произошел (не заключая все тело run в блок try-catch)? Есть ли способ восстановить работу потока после того как это произошло?

35. Какие стандартные инструменты Java вы бы использовали для реализации пула потоков?
36. Что такое ThreadGroup и зачем он нужен?
37. Что такое ThreadPool и зачем он нужен?
38. Что такое ThreadPoolExecutor и зачем он нужен?
39. Что такое «атомарные типы» в Java?
40. Зачем нужен класс ThreadLocal?
41. Что такое Executor?
42. Что такое ExecutorService?
43. Зачем нужен ScheduledExecutorService?

## 2. Практические вопросы.

1. В методе main создать фиксированный пул из 5 трэдов используя класс Executors.
2. В методе main создать очередь LinkedBlockingQueue<Runnable>. В цикле добавлять в очередь 10 задач Runnable. У каждой задачи в методе run вызывать метод doExpensiveOperation с порядковым номером задачи начиная с 1.

### Типовой экзаменационный билет

#### ГОУ ВПО «ЛУГАНСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ ВЛАДИМИРА ДАЛЯ»

**Кафедра** Информационные и управляющие системы

**Семестр 7**

**Экзаменационный контроль**

**Дисциплина** «Построение распределенных систем мониторинга»

#### ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Особенности использования методов wait и notify.
2. Что используется в качестве mutex, если метод объявлен static synchronized? Можно ли создавать новые экземпляры класса, пока выполняется static synchronized метод?
3. В методе main создать фиксированный пул из 5 трэдов используя класс Executors.

**Утверждено на заседании кафедры** \_\_\_\_\_

**Протокол №** \_\_\_\_\_

**Зав. кафедрой** \_\_\_\_\_ **Преподаватель** \_\_\_\_\_

Таблица 2.3 – Критерии и шкала оценивания по оценочному средству промежуточный контроль (зачет)

| Характеристика знания предмета и ответов                                                                                                                                                                                                                                                                                                                                  | Зачеты  |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| <p>Студент глубоко и в полном объёме владеет программным материалом.</p> <p>Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.</p> | зачтено |

|                                                                                                                                                                                                                                                                                                                                   |            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.            |            |
| Студент знает только основной программный материал, допускает неточности, недостаточно чёткие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.          |            |
| Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы. | не зачтено |

### **3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков**

Требования к выполнению контрольных заданий определены Методическими указаниями для лабораторных работ по дисциплине «Построение распределенных систем мониторинга», (электронное издание) (для студентов по специальности 09.03.02 Информационные системы и технологии).

Контрольные сроки защиты лабораторных работ определены графиком учебного процесса, в частности, лабораторные работы 1-5 должны быть защищены в первой половине семестра, 6-10 до начала экзаменационной сессии.

Требования к выполнению индивидуального задания определены методическими указаниями к выполнению индивидуального задания по дисциплине «Построение распределенных систем мониторинга», направление подготовки 09.03.02 Информационные системы и технологии (для студентов дневного и заочного обучения).

Контрольный срок сдачи индивидуального задания – до начала экзаменационной сессии.

Итоговая аттестация проводится в виде зачета по билетам, содержащим 2 теоретических вопроса и одно практическое задание.

#### 4. Лист изменений и дополнений

| №<br>п/п | Виды дополнений и<br>изменений | Дата и номер протокола<br>заседания кафедры<br>(кафедр <sup>1</sup> ), на котором<br>были рассмотрены и<br>одобрены изменения и<br>дополнения | Подпись (с<br>расшифровкой)<br>заведующего кафедрой<br>(заведующих кафедрами) |
|----------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
|          |                                |                                                                                                                                               |                                                                               |
|          |                                |                                                                                                                                               |                                                                               |
|          |                                |                                                                                                                                               |                                                                               |
|          |                                |                                                                                                                                               |                                                                               |

*Примечание:*

<sup>1</sup> - для ФОС по государственной итоговой аттестации указываются реквизиты протоколов заседания кафедр и подписи заведующих кафедрами, деканов/директоров, совместно реализующих ОП

## Экспертное заключение

Представленный фонд оценочных средств (далее – ФОС) по дисциплине «Построение распределённых систем мониторинга» соответствует требованиям ФГОС ВО.

Предлагаемые формы и средства текущего и промежуточного контроля адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 09.03.02 Информационные системы и технологии.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины представлены в полном объеме.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

Разработанный и представленный для экспертизы фонд оценочных средств рекомендуется к использованию в процессе подготовки обучающихся по указанному направлению.

Председатель учебно-методической  
комиссии факультета компьютерных  
систем и информационных  
технологий



Ветрова Н. Н.