

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Порохня Андрей Александрович

Должность: И.о. директора института перспективной инженерии

Дата подписания: 19.06.2026 17:11:11

Уникальный программный ключ:

990bea3aeec48c23bd8699e48288f8d1960c806

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

И.о. директора института перспективной инженерии, кандидат технических наук, доцент

А.А. Порохня

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ Параллельное программирование

Направление подготовки	09.04.02 Информационные системы и технологии
Направленность (профиль)	Искусственный интеллект, математическое моделирование и суперкомпьютерные технологии в разработке информационных систем
Форма обучения	очная
Год начала обучения	2026
Реализуется в семестрах	1,2

Введение

1. Назначение: Оценочные материалы (оценочные средства) предназначены для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся, и представляют собой совокупность контрольно-измерительных материалов и методов их использования для измерения уровня достижения обучающимися установленных результатов обучения.

2. ФОС является приложением к программе дисциплины «Параллельное программирование»

3. Разработчик Николаев Е.И., доцент департамента цифровых, робототехнических систем и электроники

4. Проведена экспертиза ФОС.

Члены экспертной группы:

Председатель: Шагрова Г.В., профессор департамента цифровых, робототехнических систем и электроники

Члены комиссии:

Крахоткина Е.В., доцент департамента цифровых, робототехнических систем и электроники

Винокурский Д.Л., доцент департамента цифровых, робототехнических систем и электроники

Представитель организации-работодателя: А. В. Новиков, технический директор АСУТП ООО «ЕНДС-Ставрополь»

Экспертное заключение: ФОС по дисциплине «Параллельное программирование» позволяет оценить уровень сформированности компетенций. Рекомендовать к использованию в учебном процессе.

5. Срок действия ФОС определяется сроком реализации образовательной программы.

1. Описание показателей и критериев оценивания на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенций, индикаторов	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
<i>Компетенция: ПК-2</i>				
ПК-2.1: Управляет процессами разработки и сопровождения требований к системам и качеством систем, аналитическими ресурсами, выбирает, разрабатывает и проводит экспериментальную проверку работоспособности программных компонентов систем, основанных на знаниях	Вырабатывает ошибочные управленческие решения в процессе разработки требований к системам	Вырабатывает управленческие решения (с грубыми ошибками) в процессе разработки требований к системам, управляет качеством систем	Вырабатывает управленческие решения (с незначительными ошибками) в процессе разработки требований к системам, обоснованно управляет качеством систем	Правильно управляет процессами разработки и сопровождения требований к системам и качеством систем, аналитическими ресурсами. Грамотно выбирает, разрабатывает и проводит экспериментальную проверку работоспособности программных компонентов систем.
ПК-2.2: Проводит экспериментальную проверку работоспособности систем, по обеспечению требуемых критериев эффективности и качества функционирования.	Не способен проводить экспериментальную проверку работоспособности систем, основанных на знаниях	Осуществляет с грубыми ошибками экспериментальную проверку работоспособности систем, основанных на знаниях	Осуществляет с незначительными ошибками экспериментальную проверку работоспособности систем, основанных на знаниях	Обеспечивает требуемые критерии эффективности и качества работоспособности систем.
<i>Компетенция: ПК-12</i>				
ПК-12.1: Разрабатывает методику выполнения аналитических работ в контексте исследования модели объектов профессиональной деятельности на основе методов математического моделирования и искусственного интеллекта	Низкий уровень сформированных компетенций не позволяет студенту разрабатывать методику выполнения аналитических работ в контексте исследования модели объектов профессиональной деятельности на основе методов математического моделирования и искусственного интеллекта	Допускает грубые ошибки при методике выполнения аналитических работ в контексте исследования модели объектов профессиональной деятельности на основе методов математического моделирования и искусственного интеллекта	Допускает незначительные ошибки при методике выполнения аналитических работ в контексте исследования модели объектов профессиональной деятельности на основе методов математического моделирования и искусственного интеллекта	Правильно выбирает и грамотно применяет методы математического моделирования и искусственного интеллекта для разработки методики выполнения аналитических работ при исследовании моделей

				объектов профессиональной
ПК-12.2: Планирует и организует аналитические работы в информационно-технологическом проекте на основе суперкомпьютерных технологий и методов искусственного интеллекта	Допускает грубые ошибки при планировании и организации работ в информационно-технологическом проекте, что приводит к невозможности использования суперкомпьютерных технологий и методов искусственного интеллекта	Допускает грубые ошибки при планировании и организации работ в информационно-технологическом проекте на основе суперкомпьютерных технологий и методов искусственного интеллекта	Допускает незначительные ошибки при планировании и организации работ в информационно-технологическом проекте на основе суперкомпьютерных технологий и методов искусственного интеллекта	Грамотно планирует и организует аналитические работы в информационно-технологическом проекте на основе суперкомпьютерных технологий и методов искусственного интеллекта.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
		Задания закрытого типа	
1.	1, 2	Какие типы относятся к типам-делегатам? 1) Action<T> 2) Func<T> 3) List<T> 4) IEnumerable<T>	ПК-2, ПК-12
2.	1	Какой метод позволяет реализовать остановку выполнения программы и ожидание продолжения выполнения в течении заданного количества миллисекунд? 1) Thread.Sleep() 2) System.Console() 3) Math.Wait() 4) Waiting.Stop()	ПК-2, ПК-12
3.	1	Какой класс используется для представления потока в многопоточном приложении? 1) Thread 2) Task 3) MultiTask 4) Waiting 5) Action<T>	ПК-2, ПК-12
4.	2	Какое пространство имен является корневым в библиотеке .NET 1) Data 2) System 3) Object 4) Input	ПК-2, ПК-12

		5) IO	
5.	1,2,4	В каких фрагментах кода объявляются переменные? 1) <code>int a;</code> 2) <code>int b = 7;</code> 3) <code>a = 123;</code> 4) <code>string str = "Hello, World!!!";</code>	ПК-2, ПК-12
6.	3	С помощью какого зарезервированного слова происходит объявление пространства имен? 1) using 2) break 3) namespace 4) foreach 5) name	ПК-2, ПК-12
7.	1	Даны определения классов на языке C#. <pre> class Child : Parent { int b; void E() { b = a; } } public class Parent { protected int a; } class Child1 : Parent { int c; public Child1() { } } </pre> Существуют ли ошибки в данном фрагменте? 1) ошибок нет 2) переменная a является недоступной в производном классе, следовательно, присвоение b = a некорректно 3) в двух классах отсутствуют конструкторы 4) все поля классов являются закрытыми, поэтому их использование не представляется возможным	ПК-2, ПК-12

8.	2	Что такое конструктор по умолчанию в языке C#? 1) Конструктор по умолчанию представляет собой метод, который вызывается перед созданием объекта и устанавливает значения всех полей класса в предопределенные значения 2) Программист, определяя класс, может явно не реализовывать конструктор. В этом случае .NET Framework использует пустой метод в качестве конструктора по умолчанию 3) Программист определяет сколько угодно своих конструкторов, но .NET Framework будет игнорировать их, если не отключен конструктор по умолчанию	ПК-2, ПК-12
9.	4	Даны определения классов на языке C#. <pre> public class Parent { protected int a; Parent() { a = 0; } } class Child1 : Parent { int c; public Child1() { :base() } } </pre> Существуют ли ошибки в данном фрагменте? 1) ошибок нет 2) конструктор базового класса private, а производного – public 3) все поля классов являются закрытыми, поэтому их использование не представляется возможным 4) конструктор базового класса не виден из производного класса	ПК-2, ПК-12
10.	4	Даны определения классов на языке C#.	ПК-2, ПК-12

		<pre> public class Parent { protected int a; protected Parent() { a = 0; } } class Child1 : Parent { int c; public Child1() { :base() } } </pre> Существуют ли ошибки в данных фрагментах? 1) конструктор базового класса не виден из производного класса 2) конструктор базового класса private, а производного – public 3) все поля классов являются закрытыми, поэтому их использование не представляется возможным 4) ошибок нет	
11.	1	Даны определения классов на языке C#. <pre> public class Parent: Child1 { protected int a; protected Parent() { a = 0; } } class Child1 : Child { int c; } class Child : Parent { int b; void E() { b = a; } } </pre> Существуют ли ошибки в данных фрагментах? 1) классы наследуются по кругу 2) неверное использование модификатора доступа protected в классе Parent	ПК-2, ПК-12

		3) в двух классах отсутствуют конструкторы 4) все поля классов являются закрытыми, поэтому их использование не представляется возможным 5) ошибок нет	
12.	1	Сколько наследников может быть у представленного класса? <pre> public class Child : Parent { int b; void E() { b = a; } } </pre> 1) столько, сколько реализует программист (принципиальных ограничений нет) 2) количество наследников ограничивается в базовом классе Parent 3) не более 128	ПК-2, ПК-12
13.	2	В языке C# программист определил класс следующим образом) <pre> public class Parent { protected int a; protected Parent() { a = 0; } } </pre> Какой класс является базовым для данного класса? 1) System.Data 2) System.Object 3) System.IO 4) нет базового класса	ПК-2, ПК-12

14.	3	В языке C# программист определил класс следующим образом) <pre> public class Parent { protected int a; protected Parent() { a = 0; } } </pre> Как характеризуется переменная a по отношению ко всем производным классам? 1) в производных классах переменная a является недоступной как внутри класса, так и из объектной переменной 2) в производных классах переменная a является недоступной внутри класса, но доступной из объектной переменной 3) в производных классах переменная a является видимой внутри класса, но недоступной из объектной переменной 4) в производных классах переменная a является видимой как внутри класса, так и из объектной переменной	ПК-2, ПК-12
15.	3	В языке C# программист определил классы и создал их объекты следующим образом) <pre> public class Parent { protected int a; protected Parent(int pA) { a = pA; } } public class Child : Parent { int b; public Child() :base(0) { b = a; } } static void Main(string[] args) { Parent ob1 = new Parent(); Child ob2 = new Child(); } </pre> Какая ошибка была допущена? 1) некорректное использование зарезервированного слова base	ПК-2, ПК-12

		2) в конструкторе класса Child нет доступа к переменной a базового класса 3) нельзя создавать объект ob1, так как класс Parent не имеет открытого конструктора 4) нельзя создать объект базового класса, так как объявлен класс-наследник 5) ошибок нет	
16.	1	В языке C# программист определил классы и создал их объекты следующим образом) <pre> public class Parent { protected int a; protected Parent(int pA) { a = pA; } public Parent() : this(0) { } } public class Child : Parent { int b; public Child() : base() { b = a; } } static void Main(string[] args) { Parent ob1 = new Parent(); Child ob2 = new Child(); } </pre> Какая ошибка была допущена? 1) ошибок нет 2) некорректное использование зарезервированного слова base 3) в конструкторе класса Child нет доступа к переменной a базового класса 4) некорректное использование зарезервированного слова this 5) нельзя создавать объект ob1, так как класс Parent не имеет открытого конструктора	ПК-2, ПК-12
17.	1	В языке C# программист определил классы и интерфейсы следующим образом)	ПК-2, ПК-12

		<pre> public class MainObject: IString, IDraw { int data; public string GetStr() { return "Данные: " + data.ToString(); } public void Draw() { Console.WriteLine(data.ToString()); } } </pre> Какая ошибка была допущена? 1) нет ошибок 2) класс MainObject не может иметь двух базовых классов 3) некорректное использование зарезервированного слова interface 4) в классе IString метод GetStr не определен	
18.	2	На языке C# программист определил класс, интерфейсы и создал объекты следующим образом)	ПК-2, ПК-12

		<pre> public interface IDraw { void Draw(); } public interface IString { string GetStr(); } public class Parent { protected int a; } public class MainObject: IString, IDraw { int data; public string GetStr() { return "Данные: " + data.ToString(); } public void Draw() { Console.WriteLine(data.ToString()); } } static void Main(string[] args) { MainObject ob1 = new MainObject(); ob1.Draw(); Parent ob2 = new Parent(); ob2.Draw(); } </pre> Какая ошибка была допущена? 1) в классе Parent нет конструктора, поэтому объект ob2 не может быть создан 2) в классе Parent не определен метод Draw, который вызывается через объектную переменную ob2 3) класс MainObject не может иметь двух базовых классов 4) ошибок нет	
19.	2	На языке C# программист определил класс, интерфейсы и создал объекты следующим образом)	ПК-2, ПК-12

		<pre> public interface IDraw { void Draw(); } public interface IString { string GetStr(); } public class MainObject: IString, IDraw { int data; public string GetStr() { return "Данные: " + data.ToString(); } </pre> Какая ошибка была допущена? 1) класс MainObject не может иметь двух базовых классов 2) класс MainObject не реализует метод Draw, наследуемого интерфейса IDraw 3) класс MainObject содержит метод GetStr, который дублирует метод базового класса IString 4) ошибок нет	
20.	1	На языке C# программист определил класс, интерфейсы) <pre> public interface IString { string GetStr(); } public class Data1 : IString { int data; public string GetStr() { return "Данные: " + data.ToString(); } } </pre> <pre> public class Data2 : IString { string data; public string GetStr() { return data; } } </pre> Затем программист реализовал функцию Main)	ПК-2, ПК-12

		<pre>static void Main(string[] args) { IString[] mas = new IString[] { new Data1(), new Data2() }; }</pre> Какая ошибка была допущена? 1) нет ошибок 2) в массиве присутствуют элементы различного типа 3) не указан размер массива при его создании 4) тип массива, заявленный при объявлении, и тип добавляемых элементов не совпадают	
21.	2	На языке C# программист определил класс, интерфейсы) <pre>public interface IString { string GetStr(); }</pre> <pre>public class Data1 : IString { int data; public string GetStr() { return "Данные: " + data.ToString(); } }</pre> <pre>public class Data2 : IString { string data; public string GetStr() { return data; } }</pre> Затем программист реализовал функцию Main) <pre>static void Main(string[] args) { IString[] mas = new IString[] { new Data1(), new Data2() }; IString ob = new IString(); }</pre> Какая ошибка была допущена? 1) нет ошибок	ПК-2, ПК-12

		2) Невозможно создать объект интерфейса <code>ob</code> 3) в массиве присутствуют элементы различного типа 4) тип массива, заявленный при объявлении, и тип добавляемых элементов не совпадают	
22.	1	Для чего служит оператор <code>break</code> ? 1) прерывает выполнение текущей итерации цикла и цикла в целом 2) прерывает выполнение текущей итерации цикла и переходит к следующей итерации 3) прерывает выполнение текущей итерации цикла и переходит к выполнению всего цикла с начала	ПК-2, ПК-12
23.	2	Для чего служит оператор <code>continue</code> ? 1) прерывает выполнение текущей итерации цикла и цикла в целом 2) прерывает выполнение текущей итерации цикла и переходит к следующей итерации цикла 3) прерывает выполнение текущей итерации цикла и переходит к выполнению всего цикла с начала	ПК-2, ПК-12
24.	3	В переменной <code>Sum</code> по завершении выполнения фрагмента кода будет содержаться... <pre> long Sum = 0; int i; for (i = 1; i <= 1000000; i++) if (i % 3 == 0) Sum += i; </pre> 1) сумма кубов всех чисел от 1 до 1000000 2) 0 3) сумма всех чисел кратных 3 и находящихся в диапазоне от 1 до 1000000	ПК-2, ПК-12
25.		Какие СУБД являются реляционными? +: MS SQL Server	ПК-2, ПК-12

		-: MongoDB +: MySQL +: PostgreSQL	
26.		Какие СУБД являются документоориентированными? +: MongoDB -: MySQL -: PostgreSQL -: MS SQL Server	ПК-2, ПК-12
27.		В каком формате представляются документы в MongoDB? +: JSON -: XML -: XLS -: HTTP	ПК-2, ПК-12
28.		Как происходит создание базы данных при использовании консоли MongoDB? +: база данных создается при вызове функции записи документа (документов), вместе с коллекцией, содержащей документы -: база данных создается специальной командой; MongoDB не предполагает создания базы данных «на лету» -: создать базу данных можно только из клиентского приложения на языке Python	ПК-2, ПК-12
29.		При описании подхода NoSQL используют правило 5V. Какие свойства присутствуют в правиле 5V? +: скорость (velocity) +: объем (volume) -: согласованность (consistency) -: атомарность (atomicity)	ПК-2, ПК-12
30.		При описании подхода NoSQL используют правило 5V. Какие свойства присутствуют в правиле 5V? +: достоверность (veracity) +: объем (volume) -: согласованность (consistency) +: значимость (value)	ПК-2, ПК-12
31.		Какое утверждение о формате JSON корректно? +: это текстовый формат	ПК-2, ПК-12

		-: это бинарный формат -: это табличный формат -: это мультимедия-контент	
32.		Какое утверждение об объекте JSON корректно? +: это неупорядоченное множество пар имя/значение, заключённое в фигурные скобки { } -: это неупорядоченное множество пар имя/значение, заключённое в квадратные скобки [] -: это класс, определяемый на языке Javascript	ПК-2, ПК-12
33.		Какой из объектов JSON определен корректно? +: {“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} -: /*“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”*/ -: [“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”] -: class P{“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} -: это класс, определяемый на языке Javascript	ПК-2, ПК-12
34.		Какой из объектов JSON определен корректно? +: {“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} -: <--“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”--!> -: [“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”] -: class P{“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”}	ПК-2, ПК-12
35.		В консоли MongoDB необходимо выбрать базу users в качестве текущей. Какую команду консоли следует использовать? +: use users -: use database users -: using users -: select users	ПК-2, ПК-12
36.		какая команда консоли позволяет разработчику получить справку по функционалу консоли MongoDB? +: help -: use -: select -: what	ПК-2, ПК-12
37.		В консоли MongoDB требуется узнать имя текущей используемой базы данных. Какую команду надо использовать для этого? +: db.getName()	ПК-2, ПК-12

		-: db.help() -: db.use() -: db.select()	
38.		Какая команда консоли MongoDB позволяет вывести список баз данных? +: show dbs -: dbs list -: list -: print databases	ПК-2, ПК-12
39.		Какая команда консоли MongoDB позволяет вывести список коллекций текущей базы данных? +: show collections -: print collections -: collections -: db.list()	ПК-2, ПК-12
40.		Какая команда используется для запуска консоли MongoDB? +: mongo -: startmongo -: mongod -: start mongod	ПК-2, ПК-12
41.		Какая команда используется для импорта данных из внешнего источника в базу данных MongoDB? +: mongoimport -: importdata -: importmongo -: dataimport -: import	ПК-2, ПК-12
42.		Какие стратегии для работы с MongoDB доступны разработчику? +: работа с MongoDB посредством консоли +: работа с MongoDB с использованием специального графического приложения Compass -: работа с MongoDB из текстового редактора	ПК-2, ПК-12
43.		Необходимо добавить документ в коллекцию persons текущей базы данных. Какая команда является корректной? +: db.persons.insert ({"name" : "No name"}) -: db.persons ({"name" : "No name"})	ПК-2, ПК-12

		-: db.persons.create({"name" : "No name"}) -: db.insert({"name" : "No name"})	
44.		Необходимо добавить один документ в коллекцию persons текущей базы данных. Какие команды являются корректными? +: db.persons.insert({"name" : "No name"}) -: db.persons({"name" : "No name"}) -: db.persons.insertOne({"name" : "No name"}) -: db.insert({"name" : "No name"})	ПК-2, ПК-12
45.		Необходимо добавить один документ в коллекцию persons текущей базы данных. Какие команды являются корректными? +: db.persons.insert({"name" : "No name"}) -: db.persons({"name" : "No name"}) -: db.persons.set({"name" : "No name"}) -: db.insert({"name" : "No name"})	ПК-2, ПК-12
46.		Необходимо добавить несколько документов в коллекцию persons текущей базы данных. Какие команды являются корректными? +: db.persons.insert ([{"name" : "No name"}, {"name" : "Max"}]) -: db.persons.select ([{"name" : "No name"}, {"name" : "Max"}]) +: db.persons.insertMany ([{"name" : "No name"}, {"name" : "Max"}]) -: db.persons.insert ({"name" : "No name"}, {"name" : "Max"})	ПК-2, ПК-12
47.		Необходимо выбрать все документы из коллекции zoo текущей базы данных. Какую команду консоли следует использовать? +: db.zoo.find() -: db.zoo.select() -: db.zoo.findAndSelect() -: db.find()	ПК-2, ПК-12
48.		Необходимо выбрать все документы из коллекции cats текущей базы данных. Какую команду консоли следует использовать? +: db.cats.find() -: db.cats.select() -: db.cats.findAndSelect() -: db.find("cats")	ПК-2, ПК-12

49.		Необходимо выбрать все документы из коллекции person текущей базы данных. Какую команду консоли следует использовать? <pre> +: db.person.find() -: db.person.select() -: db.person.findAndSelect() -: db.find() </pre>	ПК-2, ПК-12
50.		Необходимо выбрать один документ из коллекции zoo текущей базы данных, удовлетворяющий критерию поиска. Какую команду консоли следует использовать? <pre> +: db.person.findOne({name: "Tiger"}) -: db.person.find({name: "Tiger"}) -: db.person.One({name: "Tiger"}) -: db.person.selectOne({name: "Tiger"}) </pre>	ПК-2, ПК-12
51.		Необходимо выбрать один документ из коллекции zoo текущей базы данных, удовлетворяющий критерию поиска. Какую команду консоли следует использовать? <pre> +: db.person.findOne({age: 30}) -: db.person.find({age: 30}) -: db.person.One({age: 30}) -: db.person.selectOne({age: 30}) </pre>	ПК-2, ПК-12
52.		В коллекции goods содержатся документы: <pre> {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} </pre> Были выполнены следующие команды: <pre> db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert([{name: "No name"}, {type: "B", name: "Духи"}]) db.goods.getcount() </pre> Что будет выведено в консоль? <pre> -: 6 +: Error -: 2 -: 3 -: 1 -: 5 </pre>	ПК-2, ПК-12

		-: 4	
		Задания открытого типа	ПК-2, ПК-12
1.	2	В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find({type: "Town"})?	ПК-2, ПК-12
2.	0	В коллекции places содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.places.find({name: "Town"})?	ПК-2, ПК-12
3.	1	В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find({type: "Village"})?	ПК-2, ПК-12
4.	3	В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find()?	ПК-2, ПК-12
5.	5	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"} Сколько документов вернет запрос db.goods.find()?	ПК-2, ПК-12
6.	2	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Сколько документов вернет запрос db.goods.find({type: "A"})?	ПК-2, ПК-12

		<pre>{type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос db.goods.find({ type : "B" })?	
7.	1	В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос db.goods.findOne({ type : "B" })?	ПК-2, ПК-12
8.	1	В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос db.goods.find({ type : "B" }).limit(1)?	ПК-2, ПК-12
9.	2	В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос db.goods.find({ type : "B" }).limit(3)?	ПК-2, ПК-12
10.	2	В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос db.goods.find({ type : "B" }).limit(0)?	ПК-2, ПК-12
11.	3	В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"}</pre>	ПК-2, ПК-12

		{type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Сколько документов вернет запрос db.goods.find().limit(0)?	
12.	1	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Сколько документов вернет запрос db.goods.find({ type : "B" }).limit(2)?	ПК-2, ПК-12
13.	2	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Сколько документов вернет запрос db.goods.find({ type : "A" }).limit(2)?	ПК-2, ПК-12
14.	2	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Сколько документов вернет запрос db.goods.find({ type : "A" }).limit(0)?	ПК-2, ПК-12
15.	2	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Что вернет запрос db.goods.find({ type : "A" }).count()?	ПК-2, ПК-12
16.	1	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Что вернет запрос db.goods.find({ type : "B" }).count()?	ПК-2, ПК-12
17.	3	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко", date : "07.2019"} {type: "B", name: "Спрей"} Были выполнены следующие команды:	ПК-2, ПК-12

		db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert ([{name: "No name"}, {type: "B", name: "Спрей"}]) db.goods.find({ type: "A"}).count() Что будет выведено в консоль?	
18.	2	В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Молоко"} Были выполнены следующие команды: db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert ([{name: "No name"}, {type: "C", name: "Духи"}]) db.goods.find({ name: "Молоко"}).count() Что будет выведено в консоль?	ПК-2, ПК-12

2. Описание шкалы оценивания

В рамках рейтинговой системы успеваемость студентов по каждой дисциплине оценивается в ходе текущего контроля и промежуточной аттестации. Рейтинговая система оценки знаний студентов основана на использовании совокупности контрольных мероприятий по проверке пройденного материала (контрольных точек), оптимально расположенных на всем временном интервале изучения дисциплины. Принципы рейтинговой системы оценки знаний студентов основываются на требованиях, описанных в Положении об организации образовательного процесса на основе рейтинговой системы оценки знаний студентов в ФГАОУ ВО «СКФУ».

3. Критерии оценивания компетенций заданий открытого типа

Оценка **«отлично»** выставляется студенту, если являются верными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; представленный документ не содержит ошибок и решены все задачи. Компетенция ПК-2, ПК-12 освоена на высоком уровне.

Оценка **«хорошо»** выставляется студенту, если являются верными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; отчетный документ содержит незначительные ошибки, не влияющие на общий результат и не представлены отдельные задачи, не влияющие на общий результат. Компетенция ПК-2, ПК-12 освоена на среднем уровне.

Оценка **«удовлетворительно»** выставляется студенту, если в основном верными являются результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; отчетный документ содержит ошибки, не позволяющие оценить качество разработки; представлены не все данные и задачи решены не в полном объеме. Компетенция ПК-2, ПК-12 освоена на минимальном уровне.

Оценка **«неудовлетворительно»** выставляется студенту, если являются неверными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; представленный документ содержит много грубых ошибок, не представлены ключевые данные и задачи решены недостаточно полно. Компетенция ПК-2, ПК-12 не сформирована.

4. Критерии оценивания компетенций заданий закрытого типа

Оценка **«отлично»** выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ПК-2, ПК-12 освоена на высоком уровне.

Оценка **«хорошо»** выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ПК-2, ПК-12 освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ПК-2, ПК-12 освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ПК-2, ПК-12 не сформирована.