

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Богдалова Елена Владимировна
Должность: Проректор по образовательной деятельности
Дата подписания: 20.08.2025 13:35:52
Уникальный программный ключ:
ec85dd5a839619d48ea76b2d23dba88a9c82091a

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное бюджетное образовательное
учреждение инклюзивного высшего образования**
**«Российский государственный
университет социальных технологий»
(ФГБОУ ИВО «РГУ СоцТех»)**

УТВЕРЖДАЮ

Проректор по образовательной деятельности

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Б1.О.14 Языки и методы программирования

наименование дисциплины

01.03.02 «Прикладная математика и информатика»

шифр и наименование направления подготовки

Вычислительная математика и информационные технологии

направленность (профиль)

Содержание

1. Паспорт фонда оценочных средств.....
2. Перечень оценочных средств.....
3. Описание показателей и критериев оценивания компетенций.....
4. Методические материалы, определяющие процедуры оценивания результатов обучения, характеризующих этапы формирования компетенций.....
5. Материалы для проведения текущего контроля и промежуточной аттестации.....

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине «Языки и методы программирования»

Оценочные средства составляются в соответствии с рабочей программой дисциплины и представляют собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.), предназначенных для измерения уровня достижения обучающимися установленных результатов обучения.

Оценочные средства используются при проведении текущего контроля успеваемости и промежуточной аттестации.

Таблица 1 - Перечень компетенций, формируемых в процессе освоения дисциплины

Код компетенции	Наименование результата обучения
ОПК-1	Способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности
	ОПК-1.1. Знает основы математики, физики, вычислительной техники и программирования. ОПК-1.2. Умеет решать стандартные профессиональные задачи с применением естественнонаучных и общетехнических знаний, методов математического анализа и моделирования. ОПК-1.3. Владеет навыками теоретического и экспериментального исследования объектов профессиональной деятельности.
ОПК-2.	Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
	ОПК-2.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды решения прикладных задач. ОПК-2.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ОПК-2.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.
ПК-7	Способен к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения
	ПК-7.1. Знает теоретические основы разработки программных и алгоритмических решений в области системного и прикладного программного обеспечения; математические методы решения задач, процедурный и объектно-ориентированный подходы к разработке информационных систем; актуальные проблемы в области программирования; методы и технологии программирования; языки программирования, основы технологии модульного программирования на языках высокого уровня. ПК-7.2. Умеет применить математический метод для решения задачи; подобрать рациональную технологию программирования для решения профессиональной задачи; создавать программные продукты и алгоритмические решения в области системного и прикладного программного обеспечения.

	ПК-7.3. Владеет навыками применения математических методов для решения задач и применения стандартных алгоритмов; навыками разработки и создания алгоритмических и программных решений в области системного и прикладного программного обеспечения; навыками разработки программных приложений с использованием современных языков программирования.
--	--

Конечными результатами освоения дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование дескрипторов происходит в течение всего семестра по этапам в рамках контактной работы, включающей различные виды занятий и самостоятельной работы, с применением различных форм и методов обучения (табл.2).

Таблица 2 - Формирование компетенций в процессе изучения дисциплины:

Код компетенции	Уровень освоения компетенций	Индикаторы достижения компетенций	Вид учебных занятий ¹ , работы, формы и методы обучения, способствующие формированию и развитию компетенций ²	Контролируемые разделы и темы дисциплины ³	Оценочные средства, используемые для оценки уровня сформированности компетенции ⁴
ОПК-1		<i>Знает</i>			
	Недостаточный уровень	ОПК-1. Студент не способен самостоятельно выделять главные положения в изученном материале дисциплины. Не знает основ математики, физики, вычислительной техники и программирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Базовый уровень	ОПК-1.1. Студент усвоил основное содержание материала дисциплины, но имеет пробелы в усвоении материала. Имеет несистематизированные знания об основах математики и физики. Показывает слабое знание вычислительной техники и	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

¹ Лекционные занятия, практические занятия, лабораторные занятия, самостоятельная работа...

² Необходимо указать активные и интерактивные методы обучения (например, интерактивная лекция, работа в малых группах, методы мозгового штурма и т.д.), способствующие развитию у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств.

³ Наименование темы (раздела) берется из рабочей программы дисциплины.

⁴ Оценочное средство должно выбираться с учетом запланированных результатов освоения дисциплины, например:

«Знать» – собеседование, коллоквиум, тест...

«Уметь», «Владеть» – индивидуальный или групповой проект, кейс-задача, деловая (ролевая)

игра, портфолио...

		программирования.	аттестации, подготовка и сдача экзамена		
Средний уровень	ОПК-1.1. Студент способен самостоятельно выделять главные положения в изученном материале. Знает основы математики, физики, вычислительной техники и программирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Высокий уровень	ОПК-1.1. Студент знает, понимает, выделяет главные положения в изученном материале и способен дать краткую характеристику основным идеям проработанного материала дисциплины. Знает основы математики и физики. Показывает глубокое знание вычислительной техники и программирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
	Умеет				

	Базовый уровень	ОПК-1.2. Студент непоследовательно решает стандартные профессиональные задачи с применением общеинженерных знаний. Умеет применять методы математического анализа.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Средний уровень	ОПК-1.2. Студент умеет решать стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, но допускает незначительные ошибки. Умеет применять методы математического анализа.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Высокий уровень	ОПК-1.2. Студент умеет решать стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний. Умеет применять методы математического анализа и моделирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
		<i>Владеет</i>			
	Базовый уровень	ОПК-1.3. Студент владеет основными навыками экспериментального исследования объектов. Испытывает затруднения при	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия,	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

		использовании навыков в профессиональной деятельности.	самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена		
	Средний уровень	ОПК-1.3. Студент владеет навыками экспериментального исследования объектов. Допускает ошибки при использовании навыков в профессиональной деятельности.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Высокий уровень	ОПК-1.3. Студент владеет навыками теоретического и экспериментального исследования объектов профессиональной деятельности. Не допускает ошибок при использовании навыков в профессиональной деятельности.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 1. Основные определения Раздел 2. Создание программного кода, методы его трансляции Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
ОПК-2	Недостаточный уровень	Знает ОПК-2. Студент не способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач. Не знает основ	 Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	 Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	 Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

		программирования, принципов построения языков программирования, методов трансляции с них, визуальных сред и систем разработки			
Базовый уровень	ОПК-2.1. Студент усвоил основное содержание материала дисциплины, но имеет пробелы в усвоении материала. Имеет несистематизированные знания об основах программирования, принципах построения языков программирования, методах трансляции с них, визуальных сред и систем разработки	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Средний уровень	ОПК-2.1. Студент способен самостоятельно выделять главные положения в изученном материале. Знает основы программирования, принципов построения языков программирования, методов трансляции с них, визуальных сред и систем разработки	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Высокий уровень	ОПК-2.1. Студент знает, понимает, выделяет главные положения в изученном материале и способен дать краткую характеристику основным идеям проработанного материала дисциплины. Показывает глубокое знание и понимание основ программирования, принципов	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	

		построения языков программирования, методов трансляции с них, визуальных сред и систем разработки.	сдача экзамена		
Базовый уровень	Умеет				
	ОПК-2.2. Студент испытывает затруднения при применении языков программирования, баз данных, современных программных средств разработки информационных систем для решения прикладных задач.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Средний уровень	ОПК-2.2. Студент умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем для решения прикладных задач, но допускает незначительные ошибки.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	

	Высокий уровень	ОПК-2.2. Студент умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем для решения прикладных задач.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Базовый уровень	Владеет			
		ОПК-2.3. Студент владеет основными навыками сбора, отбора и обобщения информации, современными технологиями и средствами проектирования, разработки, тестирования ПО	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Средний уровень	ОПК-2.3. Студент владеет знаниями всего изученного материала, владеет навыками сбора, отбора и обобщения информации, современными технологиями и средствами проектирования, разработки, тестирования ПО	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

	Высокий уровень	ОПК-2.3. Студент на высоком уровне владеет концептуально-понятийным аппаратом, навыками сбора, отбора и обобщения информации, современными технологиями и средствами проектирования, разработки, тестирования ПО.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 2. Создание программного кода, методы его трансляции Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
ПК-7		<i>Знает</i>			
	Недостаточный уровень	ПК-7. Студент не способен самостоятельно выделять главные положения в изученном материале дисциплины. Не знает теоретических основ разработки программных и алгоритмических решений в области системного и прикладного программного обеспечения; математических методов решения задач; актуальных проблем в области программирования; методов и технологий программирования; языков программирования, основ технологии модульного программирования на языках высокого уровня.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Базовый уровень	ПК-7.1. Студент усвоил основное содержание материала дисциплины, но имеет пробелы в усвоении материала. Имеет несистематизированные знания о теоретических основах разработки программных и алгоритмических	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

		решений в области системного и прикладного программного обеспечения. Показывает слабое знание математических методов решения задач.	и сдача промежуточной аттестации, подготовка и сдача экзамена		
Средний уровень	ПК-7.1. Студент способен самостоятельно выделять главные положения в изученном материале. Знает теоретические основы разработки программных и алгоритмических решений в области системного и прикладного программного обеспечения. Показывает знание математических методов решения задач; актуальных проблем в области программирования; методов и технологий программирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Высокий уровень	ПК-7.1. Студент знает, понимает, выделяет главные положения в изученном материале и способен дать краткую характеристику основным идеям проработанного материала дисциплины. Знает теоретические основы разработки программных и алгоритмических решений в области системного и прикладного программного обеспечения. Показывает глубокое знание математических методов решения задач; актуальных проблем в области программирования;	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	

		методов и технологий программирования; языков программирования, основ технологии модульного программирования на языках высокого уровня.			
		Умеет			
Базовый уровень	ПК-7.2. Студент непоследовательно применяет математический метод для решения задач.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	
Средний уровень	ПК-7.2. Студент умеет применять математический метод для решения задач. Умеет подбирать рациональную технологию программирования для решения профессиональных задач.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.	

	Высокий уровень	ПК-7.2. Студент умеет применять математический метод для решения задач. Умеет подбирать рациональную технологию программирования для решения профессиональных задач; создавать программные продукты и алгоритмические решения в области системного и прикладного программного обеспечения.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
		<i>Владеет</i>			
	Базовый уровень	ПК-7.3. Студент владеет основными навыками применения математических методов для решения задач. Испытывает затруднения при применении стандартных алгоритмов.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
	Средний уровень	ПК-7.3. Студент владеет навыками применения математических методов для решения задач и применения стандартных алгоритмов Допускает ошибки при разработке и создании алгоритмических и программных решений в области системного и прикладного программного обеспечения.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.

	Высокий уровень	ПК-7.3. Студент владеет навыками применения математических методов для решения задач и применения стандартных алгоритмов; навыками разработки и создания алгоритмических и программных решений в области системного и прикладного программного обеспечения; навыками разработки программных приложений с использованием современных языков программирования.	Лекционные и практические занятия, работа в малых группах, интерактивная лекция, дискуссия, самостоятельная работа обучающихся, подготовка и сдача промежуточной аттестации, подготовка и сдача экзамена	Раздел 3. Программирование Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм Раздел 5. Построение объектно-ориентированных программ Раздел 6. Разработка интерфейса	Текущий контроль – устный опрос, тестирование, разноуровневые задачи.
--	-----------------	--	--	--	---

2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ⁵

Таблица 3

№	Наименование оценочного средства	Характеристика оценочного средства	Представление оценочного средства в ФОС
1	Устный опрос	Средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде собеседования преподавателя с обучающимися.	Вопросы по темам/разделам дисциплины
2	Разноуровневые задачи	Различают задачи и задания: а) репродуктивного уровня, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умение правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины; б) реконструктивного уровня, позволяющие оценивать и диагностировать умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей; в) творческого уровня, позволяющие оценивать и диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения.	Комплект разноуровневых задач (заданий)
3	Тестирование	Средство, позволяющее оценить уровень знаний обучающегося путем выбора им одного из нескольких вариантов ответов на поставленный вопрос. Возможно использование тестовых вопросов, предусматривающих ввод обучающимся короткого и однозначного ответа на поставленный вопрос.	Тестовые задания
4	Зачет		Вопросы к зачету
5	Курсовая работа		Темы курсовых работ

⁵ Указываются оценочные средства, применяемые в ходе реализации рабочей программы данной дисциплины.

6	Экзамен		Вопросы к экзамену
---	---------	--	--------------------

3. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Оценивание результатов обучения по дисциплине Языки и методы программирования осуществляется в соответствии с Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся.

Предусмотрены следующие виды контроля: текущий контроль (осуществление контроля всех видов аудиторной и внеаудиторной деятельности обучающегося с целью получения первичной информации о ходе усвоения отдельных элементов содержания дисциплины) и промежуточная аттестация (оценивается уровень и качество подготовки по дисциплине в целом).

Показатели и критерии оценивания компетенций, формируемых в процессе освоения данной дисциплины, описаны в табл. 4.
Таблица 4.

Код компетенции	Уровень освоения компетенции	Индикаторы достижения компетенции	Критерии оценивания результатов обучения
ОПК-1 ОПК-2 ПК-7		Знает	
	Недостаточный уровень Оценка «незачтено», «неудовлетворительно»	ОПК-1.1. ОПК-2.1. ПК-7.1.	Не знает значительной части материала курса, не способен самостоятельно выделять главные положения в изученном материале дисциплины
	Базовый уровень Оценка, «зачтено», «удовлетворительно»	ОПК-1.1. ОПК-2.1. ПК-7.1.	Знает не менее 50 % основного материала курса, однако испытывает затруднения в его применении
	Средний уровень Оценка «зачтено», «хорошо»	ОПК-1.1. ОПК-2.1. ПК-7.1.	Знает основную часть материала курса, способен применить изученный материал на практике, испытывает незначительные затруднения в решении задач
	Высокий уровень Оценка «зачтено», «отлично»	ОПК-1.1. ОПК-2.1. ПК-7.1.	Показывает глубокое знание и понимание материала, способен применить изученный материал на практике
		Умеет	
	Базовый уровень	ОПК-1.2. ОПК-2.2. ПК-7.2.	Умеет воспроизвести не менее 50 % основного материала курса, однако испытывает затруднения при решении практических задач
	Средний уровень	ОПК-1.2. ОПК-2.2.	Умеет решать стандартные профессиональные задачи с применением полученных знаний, испытывает незначительные затруднения в решении задач

		<i>ПК-7.2.</i>	
	Высокий уровень	<i>ОПК-1.2. ОПК-2.2. ПК-7.2.</i>	<i>Умеет решать стандартные профессиональные задачи с применением полученных знаний, показывает глубокое знание и понимание материала, способен решить задачу при изменении формулировки</i>
		Владеет	
	Базовый уровень	<i>ОПК-1.3. ОПК-2.3. ПК-7.3.</i>	<i>Владеет навыками теоретического и экспериментального исследования объектов профессиональной деятельности, усвоил основное содержание материала дисциплины, но имеет пробелы в усвоении материала. Имеет несистематизированные знания основных разделов дисциплины.</i>
	Средний уровень	<i>ОПК-1.3. ОПК-2.3. ПК-7.3.</i>	<i>Владеет навыками теоретического и экспериментального исследования объектов профессиональной деятельности, способен самостоятельно выделять главные положения в изученном материале. Испытывает незначительные затруднения в решении задач.</i>
	Высокий уровень	<i>ОПК-1.3. ОПК-2.3. ПК-7.3.</i>	<i>Свободно владеет навыками теоретического и экспериментального исследования, показывает глубокое знание и понимание изученного материала</i>

4. Методические материалы, определяющие процедуры оценивания результатов обучения

Задания в форме устного опроса:

Устный опрос используется для текущего контроля успеваемости обучающихся по дисциплине в качестве проверки результатов освоения терминологии. Каждому студенту выдается свой собственный, узко сформулированный вопрос. Ответ должен быть четким и кратким, содержащим все основные характеристики описываемого понятия, института, категории.

Задания в форме практических работ. Комплект разноуровневых задач (заданий)

Практическая работа представляет собой контрольное мероприятие по учебному материалу каждой темы (раздела) дисциплины, состоящее в индивидуальном выполнении обучающимся практических заданий для оценки полученных знаний, умений и владений компетенциями, формируемыми по данной дисциплине.

Выполнение практических работ является средством текущего контроля успеваемости обучающихся по дисциплине и может включать в себя следующие типы заданий: задания типового вида и задания творческого характера, по результатам выполнения практических заданий обучающие оформляют отчеты, содержащие анализ полученных результатов и выводы.

Задания в форме тестирования

Тест представляет собой контрольное мероприятие по учебному материалу каждой темы (раздела) дисциплины, состоящее в выполнении обучающимся системы стандартизированных заданий, которая позволяет автоматизировать процедуру измерения уровня знаний и умений обучающегося.

Тестирование является средством текущего контроля успеваемости обучающихся по дисциплине и может включать в себя следующие типы заданий: задание с единственным выбором ответа из предложенных вариантов, задание на определение верных и неверных суждений; задание с множественным выбором ответов.

В каждом задании необходимо выбрать все правильные ответы.

5. Материалы для проведения текущего контроля и промежуточной аттестации

Задания в форме устного опроса

Раздел 1. Основные определения

1. Общие сведения о архитектурах ЭВМ.
2. Представление данных в ЭВМ.
3. Понятие единицы информации.
4. Понятие о типах данных.
5. Понятие о языках программирования.

Раздел 2. Создание программного кода, методы его трансляции

1. Настройка операционной системы для работы с компилятором.
2. Загрузка среды разработчика.
3. Выбор компилятора.
4. Понятие проекта, компиляция.
5. Сборка и запуск программ.

Раздел 3. Программирование

1. Язык программирования C++;
2. Структура C++;
3. Синтаксис C++;
4. Операторы C++;
5. Структурный подход к программированию;
6. Заголовки и функции;
7. Понятие объектно-ориентированного программирования.

Раздел 4. Классы. Инкапсуляция, наследственность, полиморфизм

1. Операторы, операции и их перегрузка;
2. Область видимости данных;
3. Приватные и общие данные;
4. Виртуальные функции;
5. Понятие класса;
6. Базовые и производные классы.

Раздел 5. Построение объектно-ориентированных программ

1. Построение простейших классов.
2. Классы матрица и вектор.
3. Перегрузка операций.
4. Конструирование методов класса.

Раздел 6. Разработка интерфейса

1. Библиотека классов MFC;
2. Классы Документ и вид.
3. Понятие SDI интерфейс;
4. Понятие MDI интерфейс.

Контролируемые компетенции: ОПК-1, ОПК-2, ПК-7.

Оценка компетенций осуществляется в соответствии с таблицей 4.

Задания в форме практических работ. Комплект разноуровневых задач (заданий)

- 1) Написать программу для создания одномерного массива из десяти элементов. Заполнить его значениями, равными индексу элемента массива.
- 2) Изменить написанную программу так, чтобы пользователь мог вводить значения элементов массива с клавиатуры.
- 3) Вывести заполненный с клавиатуры массив на экран.
- 4) Написать программу для нахождения суммы элементов массива.
- 5) Написать программу для сортировки элементов массива методом пузырька.
- 6) Написать программу, инициализирующую два массива размерностями 4x4 (значения вводит пользователь с клавиатуры).
- 7) Написать программу для вывода введенных элементов массива на экран.

8) Написать программу для нахождения суммы элементов введенных массивов (согласно правилам сложения матриц).

9) Написать программу для нахождения произведения элементов массивов (согласно правилу умножения двух матриц).

10) Изменить программу так, чтобы пользователь мог выбрать желаемое действие с массивами (сложение или умножение).

Контролируемые компетенции: ОПК-1, ОПК-2, ПК-7.

Оценка компетенций осуществляется в соответствии с таблицей 4.

Тестирование

1. Функция вычисляет произведение двух чисел. Исходные данные вводятся с клавиатуры. Какие проверки целесообразно ввести в программе:

- а) проверка, что исходные данные являются числами +
- б) проверки не нужны, все возможные ошибки отловит компилятор
- в) проверка исходных данных на равенство нулю

2. Для чего предназначен оператор namespace:

- а) для использования классов, переменных и функций из других модулей программы без использования заголовочных файлов
- б) для заключения в группу объявлений классов, переменных и функций в отдельный контекст со своим именем +
- в) для заключения в группу объявлений классов, переменных и функций для использования только в текущем модуле

3. Какой из компонентов может входить в интегрированную среду программирования:

- а) наладчик
- б) доводчик
- в) отладчик +

4. Какой из компонентов может входить в интегрированную среду программирования:

- а) текстовый редактор +
- б) текстовый директор
- в) текстовый модератор

5. Какой из компонентов может входить в интегрированную среду программирования:

- а) регулятор
- б) доминатор
- в) компилятор +

6. Если определена операция вычитания для двух объектов класса А, а операция преобразования к int не определена, что будет вызвано при:

А a1,a2,a3=5;

a3 = a1 – a2;

- а) только операция вычитания
- б) произойдет ошибка +
- в) преобразование к целому

7. Какой из наборов перечисляемых значений записан правильно:

- а) enum { a, b = 3, c = 4, 3 };
- б) enum { a, b, 3, 4 };
- в) enum {a, b = 3, c, d };

8. В чем различие использования следующих выражений #include <...> и #include «...»:

- а) различие заключается в методе поиска препроцессором включаемого файла +
- б) в различии использования заголовочных и исходных файлов
- в) нет различий

9. Чему будет равен результат вычисления выражения: int d=5; bool b = true, c; c = (!b||(d>3)):

- а) Ошибка компилятора
- б) false
- в) true +

10. Если в арифметическом выражении участвуют целый и вещественный операнды, то:

- а) ошибка компиляции
- б) целый тип приводится к вещественному +
- в) вещественный тип приводится к целому

11. Укажите в каком выражении произойдет потеря точности:

- а) int i; float x = 2.134, y = 3.14; i = x/y; +
- б) short i = 0x3; float x = 2.7, v; v = i + x;
- в) float M = 235.2; double Z = 3; Z *= M;

12. Если после выражения стоит точка с запятой, то:

- а) выражение вычисляется, а его значение запоминается в специальной переменной, которую можно использовать в следующем операторе
- б) это оператор-выражение, действие которого заключается в вычислении выражения +
- в) выражение вычисляется только если первой стоит операция присваивания

13. Что из себя представляет динамическое выделение памяти:

- а) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится вручную +
- б) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится автоматически после завершения программы
- в) память под объект (переменную) выделяется каждый раз при обращении к переменной

14. Отметьте истинное высказывание:

- а) переменная инициализируется, потом объявляется
- б) переменная объявляется, потом инициализируется и изменяется
- в) переменная объявляется, потом изменяется +

15. Какие операции поддаются перегрузке:

- а) унарные и бинарные +
- б) только бинарные
- в) только унарные

16. Переменная типа signed char может принимать значения:

- а) только символов английского алфавита, цифр и символа подчеркивания
- б) из первой половины кодовой таблицы +
- в) только из алфавита языка C++

17. Переменная типа signed char может принимать значения:

- а) только из алфавита языка C++
- б) только символов английского алфавита, цифр и символа подчеркивания
- в) от -128 до 127 +

18. В переменной типа unsigned char можно хранить число:

- а) -213
- б) 213 +
- в) 1213

19. В переменной типа unsigned char можно хранить число:

- а) -13
- б) 1213
- в) 13 +

20. Чему равно числовое значение выражения $e/2 * a - \text{abs}(e) * 1e0$ при $e = 4$, $a = 2$:

- а) 3
- б) 0 +
- в) 1

1.	а
2.	а
3.	в
4.	б
5.	в
6.	в
7.	в
8.	в
9.	в
10.	а
11.	в
12.	в
13.	а
14.	а
15.	в
16.	в
17.	б
18.	б
19.	в

Контролируемые компетенции: ОПК-1, ОПК-2, ПК-7.

Оценка компетенций осуществляется в соответствии с таблицей 4.

Вопросы к зачету

1. ОС UNIX. История создания и классификация.
2. ОС LINUX. История создания и классификация.
3. ОС WINDOWS. История создания, классификация.
4. Переменные окружения и настройка ОС.
5. Командный интерпретатор Bash.
6. Способы инсталляции ОС.
7. Операционная система Solaris, администрирование и настройка
8. Операционная система SUSE, администрирование и настройка.
9. Операционная система MAC OS, администрирование и настройка.
10. Операционные системы Windows после ОС Vista, администрирование и настройка.
1. Понятие визуальной среды программирования.
2. Подсистема POSIX OC Windows и ее активация.
3. Межплатформенная среда разработки Netbeans.
4. Методы инсталляции, настройки и использования NetBeans.
5. Визуальные среды Visual Studio, инсталляция, настройка, использование.
6. Компиляторы FORTRAN/C++, инсталляция, настройка, опции.
7. Проекты: общие понятия, настройки и портбельность проектов для различных систем и сред.
8. Библиотеки. Библиотеки статические и динамические, принципы организации и создания библиотек.
9. Математическая библиотека Numerical Recipes: разделы, подключение, компоновка.
10. Математическая библиотека NAG: разделы, подключение, компоновка.
11. Математическая библиотека IMSL: разделы, подключение, компоновка.
12. Математическая библиотека SPP: разделы, подключение, компоновка.
13. Графический пакет DISLIN.
14. Отладка и подключение DISLIN. Основные примеры.
15. Использование интерфейсов DISLIN при разработке программ.
16. Использование библиотек и функций DISLIN, сопровождение программ.
17. Инсталляторы и их использование.
18. Целевой процессор, методы оптимизации вычислений.
19. Обработка исключительных ситуаций.
20. Основные функциональные разновидности систем разработки программ. Особенности их использования в различных ОС.

Вопросы к экзамену

1. Компиляция, сборка и запуск программ. Основные этапы построения приложений.
2. Основные типы компьютерных архитектур; понятие кода и его оптимизации для конкретной архитектуры процессора.
3. Основные типы данных и их производные. Понятие машинной числовой оси, машинного нуля, минимально и максимально представимого числа.
4. Операторы и операции. Функции-операторы. Правила записи кода, комментарии и перенос.
5. Переключатели и операции условия: вычисляемые и исполняемые.

6. Общие правила работы с символьными данными и текстовыми строками, методами их инициализации.
7. Общие правила работы с вещественными, целыми и логическими данными, методами их инициализации.
8. Особенности распределения данных в памяти: механизмы организации и взаимодействия статической и автоматической (динамической) памяти.
9. Структуры, функции, подпрограммы, их типизация и методы объявления. Препроцессор, модуль и отображение области видимости.
10. Файлы и методы доступа к ним. Заголовочные и дисковые файлы, методы организации и подключения файлов.
11. Организация массивов и методы работы с ними. Статические и динамические массивы, инициализация, индексация и передача массивов.
12. Встроенные функции для работы с массивами.
13. Алгоритмизация вычислений: параллельные и последовательные алгоритмы.
14. Итерационные алгоритмы.
15. Алгоритмы сортировки и выбора, особенности построения вычислительных алгоритмов.
16. Общие понятия о параллельных вычислениях, функции SIMD и MIMD.
17. Общее понятие об объектно-ориентированном программировании: инкапсуляция наследование и полиморфизм.
18. Общие понятия и методы организации класса и объекта класса.
19. Интерфейс: особенности разработки и применения.
20. Виртуализация и портабельность приложений.
21. Архитектуры ЭВМ, представление данных в ЭВМ, понятие единицы информации,
22. Типы данных и языки программирования.
23. Настройка операционной системы для работы с компилятором.
24. Загрузка среды разработчика и выбор компилятора.
25. Понятие проекта, компиляция, сборка и запуск программ.
26. Язык программирования C++: структура, синтаксис, операторы.
27. Структурный подход к программированию. Заголовки и функции.
28. Объектно-ориентированного программирование
29. Операторы, операции и их перегрузка.
30. Область видимости данных: приватные и общие данные.
31. Виртуальные функции.
32. Понятие класса: базовый и производные классы.
33. Построение простейших классов. Классы матрица и вектор.
34. Перегрузка операций и конструирование методов класса.
35. Библиотека классов MFC. Классы Документ и Вид.
36. Понятие SDI, MDI интерфейсов. Ресурсы.

Контролируемые компетенции: ОПК-1, ОПК-2, ПК-7.

Оценка компетенций осуществляется в соответствии с таблицей 4.