

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лиодинович
Должность: Ректор
Дата подписания: 17.02.2026 16:42:28
Уникальный программный код:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Приложение А

Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Дагестанский государственный технический университет»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине «Объектно-ориентированное программирование»

Уровень образования

Бакалавриат

(бакалавриат/магистратура/специалитет)

Направление подготовки

бакалавриата/магистратуры/специальности 09.03.01 «Информатика и вычислительная техника»
(код, наименование направления подготовки/специальности)

Профиль направления

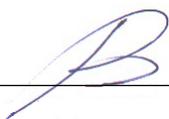
ВМКСС

(наименование)

подготовки/специализация

Разработчик

подпись

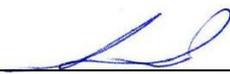


Магомедов И.А., к.т.н., доцент

Фонд оценочных средств обсужден на заседании кафедры УиИТСиВТ
« 06 » 09 2019 г., протокол № 1

Зав. кафедрой

подпись



Асланов Т.Г., к.т.н.

г. Махачкала 2019

СОДЕРЖАНИЕ

1.	Область применения, цели и задачи фонда оценочных средств.....	19
2.	Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля).....	20
2.1.	Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП.....	20
2.1.1.	Этапы формирования компетенций.....	21
2.2.	Показатели уровней сформированности компетенций на этапах их формирования, описание шкал оценивания.....	23
2.2.1.	Показатели уровней сформированности компетенций на этапах их формирования.....	23
2.2.2.	Описание шкал оценивания.....	24
3.	Типовые контрольные задания, иные материалы и методические рекомендации, необходимые для оценки сформированности компетенций в процессе освоения ОПОП.....	25
3.1.	Задания и вопросы для входного контроля.....	25
3.2.	Оценочные средства и критерии сформированности компетенций.....	25
3.3.	Задания для промежуточной аттестации (зачета и (или) экзамена).....	28

1. Область применения, цели и задачи фонда оценочных средств

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы дисциплины «Объектно-ориентированное программирование» и предназначен для контроля и оценки образовательных достижений обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данной дисциплины.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС ВО по направлению подготовки/специальности **09.03.01 Информатика и вычислительная техника.**

Рабочей программой дисциплины «Объектно-ориентированное программирование» предусмотрено формирование следующих компетенций:

ПК-6 - Способен обосновывать и принимать проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности;

ПК-16 - Способен разрабатывать компоненты программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.

2. Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля)

Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля), и используемые оценочные средства приведены в таблице 1.

2.1. Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП

		Таблица 1	
Код и наименование формируемой компетенции	Код и наименование индикатора достижения формируемой компетенции	Критерии оценивания	Наименование контролируемых разделов и тем ¹
ПК-6. Способен обосновывать и принимать проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности	ПК-6.1.1. Знает методы и формы принятия проектных решений.	Студент должен знать методы и формы принятия проектных решений.	Темы №1-17
	ПК-6.2.1. Умеет обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.	Студент должен уметь обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.	Темы №1-17
	ПК-6.3.1. Владеет навыками постановки и выполнения экспериментов по проверке их корректности и эффективности.	Студент должен владеть навыками постановки и выполнения экспериментов по проверке их корректности и эффективности.	Темы №1-17
ПК-16 - Способен разрабатывать компоненты программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования	ПК-16.1.1. Знает методы разработки программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Студент должен знать методы разработки компонент программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Темы №1-17
	ПК-16.2.1. Умеет разрабатывать компоненты программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Студент должен уметь разрабатывать компоненты программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Темы №1-17
	ПК-16.3.1. Владеет навыками разработки компонент программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Студент должен владеть навыками разработки компонент программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Темы №1-17

¹ Наименования разделов и тем должен соответствовать рабочей программе дисциплины.

2.1.1.1. Этапы формирования компетенций

Сформированность компетенций по дисциплине «Объектно-ориентированное программирование» определяется на следующих этапах:

1. Этап текущих аттестаций
2. Этап промежуточных аттестаций

Код и наименование формируемой компетенции	Этапы формирования компетенции						Этап промежуточной аттестации
	Этап текущих аттестаций						
	1-5 неделя	6-10 неделя	11-15 неделя	1-17 неделя	18-20 неделя		
1	Код и наименование формируемой компетенции	Текущая аттестация №1	Текущая аттестация №2	Текущая аттестация №3	СРС	КР/КП	Промежуточная аттестация
		2	3	4	5	6	
ПК-6. Способен обобщать и принимать проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности	ПК-6.1.1. Знает методы и формы принятия проектных решений.	Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет
		Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет
		Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет
ПК-16 - Способен разрабатывать компоненты программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования	ПК-16.1.1. Знает методы разработки компонент программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет
		Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет
		Кон-трольная работа №1	Кон-трольная работа №2	Кон-трольная работа №3	Кон-трольная работа №1, 2, 3	-	Зачет

	ПК-16.3.1. Владеет навыками разработки компонент программно-аппаратных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	Кон- трольная работа №1	Кон- трольная работа №2	Кон- трольная работа №3	Кон- трольная работа №1, 2, 3	-	Зачет
--	--	----------------------------------	----------------------------------	----------------------------------	--	---	-------

СРС – самостоятельная работа студентов;

КР – курсовая работа;

КП – курсовой проект.

2.2. Показатели уровней сформированности компетенций на этапах их формирования, описание шкал оценивания

2.2.1. Показатели уровней сформированности компетенций на этапах их формирования

Результатом освоения дисциплины «Объектно-ориентированное программирование» является установление одного из уровней сформированности компетенций: высокий, повышенный, базовый, низкий.

Таблица 3

Уровень	Универсальные компетенции	Общепрофессиональные/профессиональные компетенции
Высокий (оценка «отлично», «зачтено»)	Сформированы четкие системные знания и представления по дисциплине. Ответы на вопросы оценочных средств полные и верные. Даны развернутые ответы на дополнительные вопросы. Обучающимся продемонстрирован высокий уровень освоения компетенции	Обучающимся усвоена взаимосвязь основных понятий дисциплины, в том числе для решения профессиональных задач. Ответы на вопросы оценочных средств самостоятельны, исчерпывающие, содержание вопроса/задания оценочного средства раскрыто полно, профессионально, грамотно. Даны ответы на дополнительные вопросы. Обучающимся продемонстрирован высокий уровень освоения компетенции
Повышенный (оценка «хорошо», «зачтено»)	Знания и представления по дисциплине сформированы на повышенном уровне. В ответах на вопросы/задания оценочных средств изложено понимание вопроса, дано достаточно подробное описание ответа, приведены и раскрыты в тезисной форме основные понятия. Ответ отражает полное знание материала, а также наличие, с небольшими пробелами, умений и навыков по изучаемой дисциплине. Допустимы единичные негрубые ошибки. Обучающимся продемонстрирован повышенный уровень освоения компетенции	Сформированы в целом системные знания и представления по дисциплине. Ответы на вопросы оценочных средств полные, грамотные. Продемонстрирован повышенный уровень владения практическими умениями и навыками. Допустимы единичные негрубые ошибки по ходу ответа, в применении умений и навыков
Базовый (оценка «удовлетворительно», «зачтено»)	Ответ отражает теоретические знания основного материала дисциплины в объеме, необходимом для дальнейшего освоения ОПОП. Обучающийся допускает неточности в ответе, но обладает необходимыми знаниями для их устранения. Обучающимся продемонстрирован базовый уровень освоения компетенции	Обучающийся владеет знаниями основного материала на базовом уровне. Ответы на вопросы оценочных средств неполные, допущены существенные ошибки. Продемонстрирован базовый уровень владения практическими умениями и навыками, соответствующий минимально необходимому уровню для решения профессиональных задач
Низкий (оценка «неудовлетворительно», «не зачтено»)	Демонстрирует полное отсутствие теоретических знаний материала дисциплины, отсутствие практических умений и навыков	

Показатели уровней сформированности компетенций могут быть изменены, дополнены и адаптированы к конкретной рабочей программе дисциплины.

2.2.2. Описание шкал оценивания

В ФГБОУ ВО «ДГТУ» внедрена модульно-рейтинговая система оценки учебной деятельности студентов. В соответствии с этой системой применяются пятибалльная, двадцатибалльная и стобальная шкалы знаний, умений, навыков.

Шкалы оценивания			Критерии оценивания
пятибалльная	двадцатибалльная	стобальная	
«Отлично» - 5 баллов	«Отлично» - 18-20 баллов	«Отлично» - 85 – 100 баллов	Показывает высокий уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> - продемонстрирует глубокое и прочное усвоение материала; - исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; - правильно формирует определения; - демонстрирует умения самостоятельной работы с нормативно-правовой литературой; - умеет делать выводы по излагаемому материалу.
«Хорошо» - 4 балла	«Хорошо» - 15 - 17 баллов	«Хорошо» - 70 - 84 баллов	Показывает достаточный уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> - демонстрирует достаточно полное знание материала, основных теоретических положений; - достаточно последовательно, грамотно логически стройно излагает материал; - демонстрирует умения ориентироваться в нормальной литературе; - умеет делать достаточно обоснованные выводы по излагаемому материалу.
«Удовлетворительно» - 3 балла	«Удовлетворительно» - 12 - 14 баллов	«Удовлетворительно» - 56 – 69 баллов	Показывает пороговый уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> - демонстрирует общее знание изучаемого материала; - испытывает серьезные затруднения при ответах на дополнительные вопросы; - знает основную рекомендуемую литературу; - умеет строить ответ в соответствии со структурой излагаемого материала.
«Неудовлетворительно» - 2 балла	«Неудовлетворительно» - 1-11 баллов	«Неудовлетворительно» - 1-55 баллов	Ставится в случае: <ul style="list-style-type: none"> - незнания значительной части программного материала; - невладения понятийным аппаратом дисциплины; - допущения существенных ошибок при изложении учебного материала; - неумение строить ответ в соответствии со структурой излагаемого вопроса; - неумение делать выводы по излагаемому материалу.

3. Типовые контрольные задания, иные материалы и методические рекомендации, необходимые для оценки сформированности компетенций в процессе освоения ОПОП

3.1. Задания и вопросы для входного контроля

1. Формат программы на языке C++.
2. Директивы препроцессора.
3. Основные типы данных в C++.
4. Объявление переменных и констант в C++.
5. Операции C++.
6. Управляющие конструкции C++.
7. Объявление функций в C++. Способы передачи параметров функциям.
8. Составные типы данных в C++: массивы, структуры и объединения.
9. Указатели в C++. Работа с массивами и структурами с помощью указателей.
10. Динамическое распределение памяти.

3.2. Оценочные средства и критерии сформированности компетенций

Контрольная работа №1 по темам «Составные типы данных в C++. Динамическое распределение памяти», «ООП как направление развития теории и практики создания ПО», «Понятие класса и объекта в C++. Конструкторы и деструкторы в C++».

Комплект заданий для контрольной работы

- Время выполнения 90 мин.
- Количество вариантов контрольной работы - 6.
- Количество заданий в каждом варианте контрольной работы - 3.
- Форма работы – самостоятельная, индивидуальная.

Вариант 1

Задание 1. Анализ причин сложности современных программных систем. Основы структурного и объектного подходов к программированию. Преимущества и недостатки объектного подхода.

Задание 2. Уровни доступа к членам класса.

Задание 3. Конструктор с параметрами. Пример программы.

Вариант 2

Задание 1. Основные принципы ООП: абстракция и инкапсуляция. Преимущества инкапсуляции.

Задание 2. Объявление объектов. Создание массивов объектов. Пример программы.

Задание 3. Конструктор со списком инициализаторов. Пример программы.

Вариант 3

Задание 1. Понятие класса и объекта в ООП. Состояние и поведение объекта.

Задание 2. Указатели на объекты класса. Динамические объекты и массивы объектов. Пример программы.

Задание 3. Конструктор копирования. Поверхностное и глубинное копирование. Пример программы.

Вариант 4

Задание 1. Суть наследования и его виды.

Задание 2. Синтаксис и семантика конструктора. Указатель this. Пример программы.

Задание 3. Конструктор копирования. Поверхностное и глубинное копирование. Пример программы.

Вариант 5

Задание 1. Мономорфизм и полиморфизм.

Задание 2. Свойства конструкторов.

Задание 3. Синтаксис и семантика деструктора. Пример программы.

Вариант 6

Задание 1. Структура класса C++. Пример программы.

Задание 2. Конструктор по умолчанию. Пример программы.

Задание 3. Свойства деструкторов.

Контрольная работа №2 по темам «Члены класса, общие для всех экземпляров. Константные данные класса в C++», «Друзья класса (friend) в C++», «Механизмы наследования в C++. Простое и множественное наследование», «Перегрузка функций в C++».

Комплект заданий для контрольной работы

- Время выполнения 90 мин.
- Количество вариантов контрольной работы - 4.
- Количество заданий в каждом варианте контрольной работы - 3.
- Форма работы – самостоятельная, индивидуальная.

Вариант 1

Задание 1. Статические переменные и статические функции класса (static). Назначение. Пример программы.

Задание 2. Методы класса, дружественные другому классу. Пример программы.

Задание 3. Виртуальные базовые классы. Пример программы.

Вариант 2

Задание 1. Константные переменные и функции класса (const). Пример программы.

Задание 2. Дружественные классы. Пример программы.

Задание 3. Замещение членов базового класса. Пример программы.

Вариант 3

Задание 1. Функции, дружественные одному классу. Пример программы.

Задание 2. Объявление классов при простом наследовании. Спецификаторы наследуемого доступа. Пример программы.

Задание 3. Назначение перегрузки функций. Пример программы.

Вариант 4

Задание 1. Функции, дружественные нескольким классам. Пример программы.

Задание 2. Конструкторы и деструкторы производных классов (очередность вызова). Пример программы.

Задание 3. Перегрузка конструкторов. Пример программы.

Контрольная работа №3 по темам ««Перегрузка операторов в C++»», «Полиморфизм и виртуальные функции в C++», «Параметризованные типы в C++. Стандартная библиотека шаблонов C++», «Классы VCL-библиотеки».

Комплект заданий для контрольной работы

- Время выполнения 90 мин.
- Количество вариантов контрольной работы - 6.
- Количество заданий в каждом варианте контрольной работы - 4.

- Форма работы – самостоятельная, индивидуальная.

Вариант 1

- Задание 1. Назначение перегрузки операторов. Правила перегрузки операторов.
- Задание 2. Механизмы раннего и позднего связывания.
- Задание 3. Синтаксис шаблонного класса. Пример программы.
- Задание 4. Ассоциативные контейнеры. Пример.

Вариант 2

- Задание 1. Перегрузка бинарных операторов. Пример программы.
- Задание 2. Виртуальные функции. Таблица виртуальных функций. Пример программы.
- Задание 3. Специализация шаблонов функций и классов. Пример программы.
- Задание 4. Итераторы. Обратные итераторы. Пример.

Вариант 3

- Задание 1. Перегрузка унарных операторов. Пример программы.
- Задание 2. Чисто виртуальные функции и абстрактные классы. Пример программы.
- Задание 3. Достоинства и недостатки шаблонов.
- Задание 4. Алгоритмы библиотеки STL. Пример.

Вариант 4

- Задание 1. Перегрузка оператора доступа к членам класса. Пример программы.
- Задание 2. Назначение виртуальных деструкторов. Пример программы.
- Задание 3. Назначение и состав библиотеки STL.
- Задание 4. Иерархия классов VCL-библиотеки.

Вариант 5

- Задание 1. Перегрузка и неоднозначность. Пример программы.
- Задание 2. Назначение шаблонных функций.
- Задание 3. Последовательные контейнеры. Пример.
- Задание 4. Классы командных кнопок, флажков и радиокнопок VCL-библиотеки.

Вариант 6

- Задание 1. Виды и реализации полиморфизма.
- Задание 2. Конкретизация шаблонной функции. Пример программы.
- Задание 3. Контейнеры – адаптеры. Пример.
- Задание 4. Обработка событий средствами VCL-библиотеки.

Критерии оценки уровня сформированности компетенций при проведении контрольной работы:

- оценка «отлично»: продемонстрировано грамотное последовательное решение задач (заданий) при правильно выбранном алгоритме. Даны верные ответы на все вопросы и условия задач (заданий). При необходимости сделаны пояснения и выводы (содержательные, достаточно полные, правильные, учитывающие специфику проблемной ситуации в задаче или с незначительными ошибками);

- оценка «хорошо»: грамотное последовательное решение задач (заданий) при правильно выбранном алгоритме. Однако, ответы на вопросы и условия задач (заданий) содержат незначительные ошибки. Пояснения и выводы отсутствуют или даны неверно;

- оценка «удовлетворительно»: обучающийся ориентируется в материале, но применяет его неверно, выбирает неправильный алгоритм решения задач (неверные исходные данные, неверная последовательность решения и др. ошибки), допускает вычислительные ошибки. Пояснения и выводы отсутствуют или даны неверно;

- оценка «неудовлетворительно»: обучающийся слабо ориентируется в материале. выбирает неправильный алгоритм решения, допускает значительное количество вычислительных ошибок. Пояснения и выводы отсутствуют.

3.3. Задания для промежуточной аттестации (зачета и (или) экзамена)

Список вопросов к зачету

1. Основы структурного и объектного подходов к программированию. Преимущества и недостатки объектного подхода.
2. Уровни доступа к членам класса.
3. Конструктор с параметрами. Пример программы.
4. Основные принципы ООП: абстракция и инкапсуляция.
5. Объявление объектов. Создание массивов объектов. Пример программы.
6. Конструктор со списком инициализаторов. Пример программы.
7. Понятие класса и объекта в ООП. Состояние и поведение объекта.
8. Указатели на объекты класса. Динамические объекты и массивы объектов. Пример программы.
9. Конструктор копирования. Поверхностное и глубинное копирование. Пример программы.
10. Синтаксис и семантика конструктора. Указатель `this`. Пример программы.
11. Конструктор копирования. Поверхностное и глубинное копирование. Пример программы.
12. Суть наследования и его виды. Полиморфизм.
13. Свойства конструкторов.
14. Синтаксис и семантика деструктора. Пример программы.
15. Структура класса C++. Пример программы.
16. Конструктор по умолчанию. Пример программы.
17. Свойства деструкторов.
18. Статические переменные и статические функции класса (`static`). Назначение. Пример программы.
19. Методы класса, дружественные другому классу. Пример программы.
20. Виртуальные базовые классы. Пример программы.
21. Константные переменные и функции класса (`const`). Пример программы.
22. Дружественные классы. Пример программы.
23. Замещение членов базового класса. Пример программы.
24. Функции, дружественные одному классу. Пример программы.
25. Объявление классов при простом наследовании. Спецификаторы наследуемого доступа. Пример программы.
26. Назначение перегрузки функций. Пример программы.
27. Функции, дружественные нескольким классам. Пример программы.
28. Конструкторы и деструкторы производных классов (очередность вызова). Пример программы.
29. Перегрузка конструкторов. Пример программы.
30. Назначение перегрузки операторов. Правила перегрузки операторов.
31. Механизмы раннего и позднего связывания.
32. Синтаксис шаблонного класса. Пример программы.
33. Ассоциативные контейнеры. Пример.
34. Перегрузка бинарных операторов. Пример программы.
35. Виртуальные функции. Таблица виртуальных функций. Пример программы.
36. Специализация шаблонов функций и классов. Пример программы.
37. Итераторы. Обратные итераторы. Пример.
38. Перегрузка унарных операторов. Пример программы.
39. Чисто виртуальные функции и абстрактные классы. Пример программы.

40. Достоинства и недостатки шаблонов.
41. Алгоритмы библиотеки STL. Пример.
42. Перегрузка оператора доступа к членам класса. Пример программы.
43. Назначение виртуальных деструкторов. Пример программы.
44. Назначение и состав библиотеки STL.
45. Перегрузка и неоднозначность. Пример программы.
46. Назначение шаблонных функций.
47. Последовательные контейнеры. Пример.
48. Виды и реализации полиморфизма.
49. Конкретизация шаблонной функции. Пример программы.
50. Контейнеры – адаптеры. Пример.
51. Иерархия классов VCL-библиотеки.
52. Объекты, компоненты, элементы управления VCL-библиотеки.
53. Классы командных кнопок, флажков и радиокнопок VCL-библиотеки.
54. Классы текстовых компонентов, списков и панелей VCL-библиотеки.
55. Классы стандартных диалогов VCL-библиотеки.
56. Обработка событий средствами VCL-библиотеки.
57. Структура приложения на C# и Java.
58. Создание классов и объектов C# и Java.
59. Методы-члены класса и виртуальные методы C# и Java.
60. Механизмы наследования C# и Java.
61. Классы библиотеки .NET Framework.
62. Приложения – диалоги библиотеки классов .NET Framework.

Зачет может быть проведен в письменной форме, а также в письменной форме с устным дополнением ответа. Зачет служит формой проверки качества усвоения семестрового учебного материала по дисциплине (модулю), практических и семинарских занятий (при отсутствии экзамена по дисциплине).

По итогам зачета, в соответствии с модульно – рейтинговой системой университета, выставляются баллы с последующим переходом по шкале баллы – оценки за зачет, выставляемый по наименованию «зачет» и «незачет».