

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лиодинович
Должность: Ректор
Дата подписания: 24.03.2026 12:45:41
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Приложение А

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине «Программирование»

Уровень образования	бакалавриат (бакалавриат/магистратура/специалитет)
Направление подготовки бакалавриата/магистратуры/специальность	09.03.01 Информатика и вычислительная техника (код, наименование направления подготовки/специальности)
Профиль направления подготовки/специализация	«Компьютерные системы и технологии» (наименование)

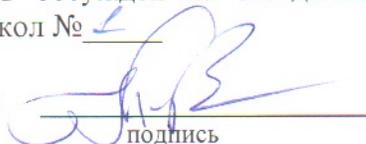
Разработчик


подпись

Айгумов Т.Г. к.э.н.
(ФИО уч. степень, уч. звание)

Фонд оценочных средств обсужден на заседании кафедры 7067uTe
«9» 09 20 21г., протокол № 1

Зав. кафедрой


подпись

Айгумов Т.Г. к.э.н.
(ФИО уч. степень, уч. звание)

г. Махачкала 2021

СОДЕРЖАНИЕ

Стр.

1.	Область применения, цели и задачи фонда оценочных средств.....	19
2.	Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля).....	19
2.1.	Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП	20
2.1.2.	Этапы формирования компетенций.....	22
2.2.	Показатели уровней сформированности компетенций на этапах их формирования, описание шкал оценивания	24
2.2.1.	Показатели уровней сформированности компетенций на этапах их Формирования.....	24
2.2.2.	Описание шкал оценивания.....	26
3.	Типовые контрольные задания, иные материалы и методические рекомендации, необходимые для оценки сформированности компетенций в процессе освоения ОПОП.....	27
3.1.	Задания и вопросы для входного контроля.....	27
3.2.	Оценочные средства и критерии сформированности компетенций	27
3.3.	Задания для промежуточной аттестации (экзамен, зачета).....	35

1. Область применения, цели и задачи фонда оценочных средств

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы дисциплины «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 09.03.01 Информатика и вычислительная техника

Рабочей программой дисциплины «Программирование» предусмотрено формирование следующих компетенций:

- 1) ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;
- 2) ОПК-3. Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;
- 3) ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;
- 4) ОПК-5. Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;
- 5) ОПК-7. Способен участвовать в настройке и наладке программно-аппаратных комплексов;
- 6) ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения;
- 7) ОПК-9. Способен осваивать методики использования программных средств для решения практических задач;

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
<p>ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности</p>	<p>ОПК-2.1. Знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.2. Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.3. Владеть: навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности</p>	<p>Знать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности Владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности</p>	<p>Темы 1-43</p>
<p>ОПК-3. Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p>	<p>ОПК-3.1. Знать: принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.2. Уметь: решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.3. Владеть: навыками подготовки обзоров, аннотаций, составления рефератов,</p>	<p>Знать: принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности Уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности Владеть навыками подготовки обзоров, аннотаций, составления рефератов, научных</p>	<p>Темы 1-43</p>

<p>ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью</p>	<p>научных докладов, публикаций и библиографии по научно-исследовательской работе с учетом требований информационной безопасности</p>	<p>докладов, публикаций и библиографии по научно-исследовательской работе с учетом требований информационной безопасности</p>	<p>Темы 1-43</p>
<p>ОПК-5. Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем</p>	<p>ОПК-4.1. Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.2. Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.3. Владеть: составлением технической документации на различных этапах жизненного цикла информационной системы</p>	<p>Знать основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы Уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы Владеть навыками составления технической документации на различных этапах жизненного цикла информационной системы</p>	<p>Темы 1-43</p>
<p>ОПК-7. Способен участвовать в настройке и наладке программно-аппаратных комплексов</p>	<p>ОПК-5.1. Знать: основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем ОПК-5.2. Уметь: выполнять параметрическую настройку информационных и автоматизированных систем ОПК-5.3. Владеть: навыками установки программного и аппаратного обеспечения информационных и автоматизированных систем</p>	<p>Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем Уметь выполнять параметрическую настройку информационных и автоматизированных систем Владеть навыками установки программного и аппаратного обеспечения информационных и автоматизированных систем</p>	<p>Темы 1-43</p>
<p>ОПК-7. Способен участвовать в настройке и наладке программно-аппаратных комплексов</p>	<p>ОПК-7.1. Знать: методы настройки, наладки программно-аппаратных комплексов ОПК-7.2. Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов ОПК-7.3. Владеть: навыками проверки работоспособности программно-аппаратных комплексов</p>	<p>Знать методы настройки, наладки программно-аппаратных комплексов Уметь анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов Владеть навыками проверки работоспособности программно-аппаратных комплексов</p>	<p>Темы 1-43</p>

<p>ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения</p>	<p>ОПК-8.1. Знать: алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения ОПК-8.2. Уметь: составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули ОПК-8.3. Владеть: языком программирования; навыками отладки и тестирования работоспособности программы</p>	<p>Знать алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения Уметь составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули Владеть языком программирования; навыками отладки и тестирования работоспособности программы</p>	<p>Темы 1-43</p>
<p>ОПК-9. Способен осваивать методики использования программных средств для решения практических задач</p>	<p>ОПК-9.1. Знать: классификацию программных средств и возможности их применения для решения практических задач ОПК-9.2. Уметь: находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые функции программных средств для решения конкретной задачи ОПК-9.3. Владеть: способами описания методики использования программного средства для решения конкретной задачи в виде документа, презентации или видеоролика</p>	<p>Знать классификацию программных средств и возможности их применения для решения практических задач Уметь находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые функции программных средств для решения конкретной задачи Владеть способами описания методики использования программного средства для решения конкретной задачи в виде документа, презентации или видеоролика</p>	<p>Темы 1-43</p>

2.1.2. Этапы формирования компетенций

Сформированность компетенций по дисциплине «Программирование» определяется на следующих этапах:

1. Этап текущих аттестаций
2. Этап промежуточных аттестаций

Таблица 2

Код и наименование формируемой компетенции	Код и наименование индикатора достижения формируемой компетенции	Этапы формирования компетенции					
		Этап текущих аттестаций			Этап промежуточной аттестации		
		1-5 неделя Текущая аттестация №1	6-10 неделя Текущая аттестация №2	11-15 неделя Текущая аттестация №3	1-17 неделя СРС	18-20 неделя Промежуточная аттестация	Вопросы для проведения зачета и экзамена
ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	ОПК-2.1. Знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	2	3	4	5	6	7
ОПК-2. Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	ОПК-2.2. Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
ОПК-3. Способен	ОПК-3.1. Знать: : принципы, методы и	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена

<p>решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p> <p>ОПК-3.2. Уметь: решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p>	<p>средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p> <p>ОПК-3.2. Уметь: решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p>	<p>Контрольная работа</p>	<p>опрос</p>	<p>КР</p>	<p>Вопросы для проведения зачета и экзамена</p>				
<p>решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p> <p>ОПК-3.3. Владеть: навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с учетом требований информационной безопасности</p>	<p>средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p> <p>ОПК-3.3. Владеть: навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с учетом требований информационной безопасности</p>	<p>Контрольная работа</p>	<p>Реферат, устный опрос</p>	<p>КР</p>	<p>Вопросы для проведения зачета и экзамена</p>				

ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	безопасности ОПК-4.1. Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.2. Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.3. Владеть: составлением технической документации на различных этапах жизненного цикла информационной системы	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
ОПК-5. Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	ОПК-5.1. Знать: основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем ОПК-5.2. Уметь: выполнять параметрическую настройку информационных и автоматизированных систем ОПК-5.3. Владеть: навыками установки программного и аппаратного обеспечения информационных и автоматизированных систем	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
ОПК-7. Способен участвовать в настройке и наладке	ОПК-7.1. Знать: методы настройки, наладки программно-аппаратных комплексов ОПК-7.2. Уметь:	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
		Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета

программно-аппаратных комплексов	анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов	работа	работа	работа	работа	устный опрос		и экзамена
ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.3. Владеть: навыками проверки работоспособности программно-аппаратных комплексов	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
	ОПК-8.1. Знать: алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
	ОПК-8.2. Уметь: составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
ОПК-9. Способен осваивать методики использования программных средств для решения практических задач	ОПК-8.3. Владеть: языком программирования; навыками отладки и тестирования работоспособности программы	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
	ОПК-9.1. Знать: классификацию программных средств и возможности их применения для решения практических задач	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена
	ОПК-9.2. Уметь: находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	Вопросы для проведения зачета и экзамена

	функции программных средств для решения конкретной задачи								
	ОПК-9.3. Владеть: способами описания методики использования программного средства для решения конкретной задачи в виде документа, презентации или видеоролика	Контрольная работа	Контрольная работа	Контрольная работа	Контрольная работа	Реферат, устный опрос	КР	<i>Вопросы для проведения зачета и экзамена</i>	

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 «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	Обучающиеся 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 незначительными 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 материи 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 баллов	«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 баллов	«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 баллов	«Неудовлетворительно» - 1-11 баллов	«Неудовлетворительно» - 1-55 баллов	Ставится в случае: <ul style="list-style-type: none"> - незнания значительной части программного материала; - не владения 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. Основные типы данных C++
3. Константы. Именованные константы
4. Операции C++. Поразрядные операции
5. Выражения. Преобразования типов данных
6. Операторы выбора
7. Операторы цикла
8. Операторы переходов
9. Массивы
10. Символьные массивы

3.2. Оценочные средства и критерии сформированности компетенций

Семестр 1

Аттестационная контрольная работа №1

1. Каковы основные этапы разработки программного обеспечения, и почему они важны?
2. Какие свойства и способы описания алгоритма вы знаете?
3. Что такое тип данных, и какие типы данных поддерживает C++?
4. Как объявляются переменные в C++? Приведите примеры объявления переменных разных типов.
5. Что такое класс памяти, и какие классы памяти существуют в C++?
6. Что такое область действия идентификатора, и как она влияет на доступ к переменным?
7. Какие свойства переменных в C++ вам известны?
8. Какие типовые алгоритмы вы можете применить в программировании на C++? Приведите примеры.
9. Что такое синтаксические ошибки в программировании, и как их избегать?
10. Что такое семантические ошибки, и как их обнаруживать и исправлять?
11. Какие типичные алгоритмы вы можете реализовать с использованием циклов в C++?
12. Какие операторы условия (if-else) вы знаете, и как они используются для управления потоком программы?
13. Какие структуры данных поддерживает C++, и как они могут быть применены для решения задач?
14. Какие функции вы можете использовать для ввода и вывода данных в C++?
15. Предоставьте пример алгоритма на C++, который использует переменные, операторы условия и циклы для выполнения определенной задачи.

Аттестационная контрольная работа №2

1. Что такое оператор объявления, и какие элементы он включает?
2. Какие типы данных могут быть объявлены с использованием оператора объявления в C++?
3. Какие основные операторы являются исполняемыми операторами в C++?
4. Каким образом можно встроить комментарии в ваш код C++?
5. Что такое условный оператор, и как он используется в программировании на C++?
6. Каков синтаксис условного оператора (if) в его краткой форме?
7. В чем заключается разница между краткой и полной формой условного оператора (if)?

8. Что такое оператор множественного выбора (switch), и как он применяется в C++?
9. Какие операторы могут быть использованы внутри оператора множественного выбора (switch)?
10. Что такое тернарный оператор, и как он отличается от обычного условного оператора (if)?
11. Что такое алгоритм линейной структуры, и как его можно использовать в программировании?
12. Какие операторы условного перехода вы можете использовать в C++? Приведите примеры их использования.
13. Какие операторы перехода (case) могут быть использованы внутри оператора множественного выбора (switch)?
14. Как вы можете составлять блок-схемы для алгоритмов и программ? Какие элементы должны быть включены в блок-схемы?
15. Предоставьте пример программы на C++, которая использует операторы условия (if и switch) и включает блок-схему для иллюстрации алгоритма.

Аттестационная контрольная работа №3

1. Что такое оператор цикла с параметром for? Какой синтаксис у него?
2. Как можно использовать оператор for для вычисления суммы произведений элементов массива?
3. Что такое цикл с предусловием while? Какой синтаксис у него?
4. В чем отличие цикла с предусловием while от цикла с постусловием do-while?
5. Как работает цикл с постусловием do-while? Приведите пример использования.
6. В каких случаях следует избегать использования безусловного перехода goto в коде?
7. Какой синтаксис оператора goto и как он может быть использован в C++?
8. Какой оператор используется для прерывания выполнения цикла и перехода к следующей итерации?
9. Что делает оператор break? Как он может быть применен в циклах?
10. Какой оператор используется для перехода к следующей итерации цикла, минуя оставшуюся часть текущей итерации?
11. Какой оператор можно использовать для завершения выполнения программы внутри функции?
12. В каких случаях целесообразно использовать оператор exit для завершения программы?
13. Каким образом можно предотвратить использование бесконечных циклов?
14. Почему использование операторов goto, break, и continue может сделать код менее читаемым и поддерживаемым?
15. Предоставьте пример программы на C++, который иллюстрирует использование циклов, операторов goto, break, continue, и exit.

Семестр 2

Аттестационная контрольная работа №1

1. Что такое указатель в C++? Как объявить указатель?
2. Какие операции над указателями можно выполнять, включая арифметические операции?
3. Какие операции отношения могут быть применены к указателям? Как они работают?
4. Что такое указатель на указатель? Как его объявить и использовать?
5. Как создать массив указателей, и в каких ситуациях это может быть полезным?
6. Как строки и указатели взаимодействуют в C++?
7. Что такое ссылка в C++? Как объявить ссылку?
8. В чем отличие указателей от ссылок, и когда следует предпочесть одно другому?
9. Как можно выделить динамическую память в C++? Какие операторы и функции используются для этого?

10. Как освобождается динамически выделенная память? Какие функции используются для этой цели?
11. Что такое динамический массив? Как его создать и освободить память после использования?
12. Какие преимущества и недостатки связаны с использованием динамических массивов по сравнению с статическими массивами?
13. Как можно создать массив указателей на строки (строки) и работать с ними?
14. Какие практические сценарии требуют использования указателей и динамической памяти в программировании на C++?
15. Предоставьте пример программы на C++, который демонстрирует работу с указателями, динамической памятью и динамическими массивами.

Аттестационная контрольная работа №2

1. Что такое прототип функции, и зачем он нужен в C++?
2. Как определяется функция в C++? Какой синтаксис используется?
3. Как осуществляется вызов функции в C++? Как передаются аргументы?
4. В чем разница между глобальными и локальными переменными? Как они влияют на область видимости?
5. Что такое классы памяти (стек и куча), и как они связаны с локальными и динамическими переменными?
6. Как передаются параметры функций в C++? Какие варианты передачи доступны?
7. Что такое значения по умолчанию для параметров функций, и как они определяются?
8. Как можно передавать массивы в функции? Какие способы передачи массивов существуют?
9. Как строки могут быть переданы в функции в C++? Какие способы передачи строк используются?
10. Что такое указатели на функции, и как они могут быть использованы?
11. Что такое рекурсивная функция, и как она отличается от обычной функции?
12. В чем состоит преимущество и недостатки использования рекурсивных функций?
13. Что такое встраиваемые функции (инлайн-функции), и как они работают?
14. Может ли функция `main()` иметь параметры? Какие параметры она может принимать, и как они используются?
15. Предоставьте пример программы на C++, который демонстрирует использование функций, передачу параметров, работу с указателями на функции и рекурсию.

Аттестационная контрольная работа №3

1. Какие символьные функции используются для проверки, является ли символ буквой или цифрой?
2. Какие символьные функции могут быть использованы для проверки, является ли символ специальным символом?
3. Какие функции предоставляются для преобразования символов в верхний и нижний регистры?
4. Какие строковые функции позволяют проверить или обработать строки, например, для удаления пробелов?
5. Как можно преобразовать строку, представляющую число, в числовое значение?
6. Какие числовые функции предоставляются для выполнения математических операций в C++?
7. Как можно сгенерировать случайное число в C++? Какие функции используются?
8. Как представляется экран в текстовом режиме с использованием библиотеки `conio.h`?

9. Какие функции из заголовочного файла `conio.h` могут быть использованы для работы с экраном в текстовом режиме?
10. Как представляется экран в графическом режиме с использованием библиотеки `graphics.h`?
11. Какие функции из заголовочного файла `graphics.h` предоставляются для работы с экраном в графическом режиме?
12. Какие задачи чаще всего решаются с использованием символьных функций?
13. В каких сценариях полезны строковые функции для обработки и проверки строк?
14. Какие типы приложений могут использовать числовые функции и функции генерации случайных чисел?
15. В каких ситуациях экран в текстовом и графическом режимах является важным элементом разработки программы?

Семестр 3

Аттестационная контрольная работа №1

1. Что такое пользовательский тип данных, и для чего используется ключевое слово `typedef`?
2. Как объявляется перечислимый тип данных в C++? Приведите синтаксис.
3. Какие операции можно выполнять над переменными перечислимого типа?
4. Как объявляется структура в C++? Какие данные она может содержать?
5. Что такое псевдонимы структур? Как они создаются?
6. Какие операции допустимы над переменными структурного типа?
7. Можно ли использовать массивы и структуры в качестве элементов структур?
8. Как объявляются массивы структур, и как можно обратиться к их элементам?
9. Как работают указатели на структуры в C++? Как обратиться к полям структуры с использованием указателей?
10. Какие операции можно выполнять над указателями на структуры?
11. Как объявляется объединение (`union`) в C++? В чем заключается особенность размещения объединения в памяти?
12. Какие операции допустимы с объединениями? Как они отличаются от операций со структурами?
13. Что такое битовые поля? Как они могут быть использованы?
14. Какие функции предоставляются для работы с датой и временем в C++?
15. В каких сценариях структуры, объединения и битовые поля могут быть полезными при разработке программ на C++?

Аттестационная контрольная работа №2

1. Что такое самоссылочная структура, и в чем заключается её особенность?
2. Как классифицируются самоссылочные структуры, и какие типы самоссылочных структур вы знаете?
3. В каких областях применяются самоссылочные структуры в программировании?
4. Как можно формировать очередь с использованием самоссылочных структур?
5. Как реализовать стек с использованием самоссылочных структур?
6. Какие операции могут быть использованы для добавления и удаления элементов в односвязных списках?
7. Как формируются двусвязные списки, и какие преимущества они имеют по сравнению с односвязными списками?
8. Какие операции можно выполнять при добавлении и удалении элементов в двусвязных списках?
9. Как можно реализовать списки с помощью массивов, и в каких случаях это может быть полезно?
10. Что такое бинарное дерево, и как оно отличается от обычных деревьев?

11. В чем заключаются особенности дерева двоичного поиска? Как выполняются поиск, вставка и удаление элементов?
12. Какие основные операции могут быть выполняемы с бинарными деревьями?
13. Как можно использовать деревья для сортировки данных, и как работает сортировка на основе деревьев?
14. Какие алгоритмы сортировки вы знаете, и в чем состоят их преимущества и недостатки?
15. В каких сценариях и для каких задач наиболее эффективно использование структур данных, изученных в этих темах?

Аттестационная контрольная работа №3

1. Какой этап включает постановку задачи и разработку функций для обслуживания базы данных?
2. Как можно сохранить базу данных на диске в C++? Какие методы используются для этого?
3. Что такое класс в C++, и как он описывается? Какие уровни доступа к элементам класса вы знаете?
4. Что такое объект в C++? Как он создается и используется?
5. Какие функции могут быть включены в состав класса, и как они вызываются?
6. Что представляет собой указатель this в классе? Как он используется?
7. Какие типы конструкторов вы можете определить для класса, и какие задачи они решают?
8. Что такое конструктор копирования, и как он отличается от других конструкторов?
9. Что выполняет деструктор в классе? Как он определяется и вызывается?
10. Как работает класс string в C++, и какие операции он поддерживает?
11. Какие конструкторы могут быть использованы для создания объектов string?
12. Как можно изменить размер строки и ее емкость в классе string?
13. Как можно сравнивать строки, и какие операции поиска поддерживаются в string?
14. Какие классификации контейнеров существуют, и в чем заключаются их различия?
15. Какие контейнеры относятся к последовательным контейнерам, и какие к ассоциативным? Какие контейнеры вы можете назвать в каждой категории?

Критерии оценки уровня сформированности компетенций при проведении контрольной работы:

- оценка «отлично»: обучающимся дан полный, развернутый ответ на поставленный вопрос; в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений. Знание по дисциплине демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей. Обучающийся владеет терминологией, способен приводить примеры, высказывает свою точку зрения с опорой на знания и опыт;

- оценка «хорошо»: обучающимся дан полный, развернутый ответ на поставленный вопрос, показано умение выделять существенные и несущественные признаки, причинно-следственные связи. Ответ логичен, выстроен, но совершены единичные ошибки. Не в полной мере владеет знаниями по всей дисциплине. Даны ответы на дополнительные, поясняющие вопросы;

- оценка «удовлетворительно»: ответ на вопрос не полный, с ошибками. Обучающийся путается в деталях, с затруднением пользуется профессиональной терминологией. Есть замечания к построению ответа, к логике и последовательности изложения. Не отвечает на дополнительные вопросы;

- оценка «неудовлетворительно»: ответ представляет собой разрозненные знания с существенными ошибками по вопросу, присутствует фрагментарность, нелогичность изложения. Обучающийся не осознает связь обсуждаемого вопроса с другими объектами дисциплины, речь неграмотная, не используется профессиональная терминология. Ответы на дополнительные вопросы не даны или неверные.

Курсовая работа Примерные темы курсовых работ

1. Разработка программы для анализа и обработки данных с использованием структур и указателей в C++.
2. Создание приложения для управления базой данных с использованием бинарных файлов и последовательных контейнеров.
3. Разработка игры на C++, используя объектно-ориентированный подход и графическую библиотеку.
4. Исследование и реализация алгоритмов сортировки и их сравнение в различных сценариях.
5. Создание консольного приложения для расчета и визуализации математических функций.
6. Разработка системы управления учетом и анализом финансовых данных на C++.
7. Реализация приложения для анализа и обработки текстовых данных, включая работу со строками и файлами.
8. Создание программы для автоматической генерации отчетов на основе данных из внешних источников.
9. Исследование и реализация алгоритмов для решения задачи поиска пути в графах.
10. Разработка приложения для визуализации и анализа данных с использованием графических библиотек.
11. Исследование и реализация алгоритмов для работы с изображениями, включая обработку и фильтрацию.
12. Создание приложения для симуляции работы производственной линии с использованием объектов и классов.
13. Разработка игры "крестики-нолики" с возможностью игры против компьютера на C++.
14. Создание системы для анализа и обработки больших объемов данных, включая параллельное выполнение.
15. Реализация программы для решения задачи криптографии с использованием алгоритмов шифрования на C++.

Требования к структуре, содержанию и оформлению курсовых работ (проектов) приводятся в методических указаниях/рекомендациях.

Критерии оценки уровня сформированности компетенций при выполнении курсовой работы/курсового проекта:

- оценка «отлично»: продемонстрировано блестящее владение проблемой исследования, материал выстроен логично, последовательно, обучающийся аргументированно отстаивает свою точку зрения. Во введении приводится обоснование выбора конкретной темы, четко определены цель и задачи работы (проекта). Использован достаточный перечень источников и литературы для методологической базы исследования. Обучающийся грамотно использует профессиональные термины, актуальные исходные данные. Проведен самостоятельный анализ (исследование) объекта. По результатам работы сделаны логичные выводы. Оформление работы соответствует методическим рекомендациям. Объем и содержание работы соответствует требованиям. На защите обучающийся исчерпывающе отвечает на все дополнительные вопросы;

- оценка «хорошо»: обучающийся демонстрирует повышенный уровень владения проблемой исследования, логично, последовательно и аргументированно отстаивает ее концептуальное содержание. Во введении содержатся небольшие неточности в формулировках цели, задач. В основной части допущены незначительные погрешности в расчетах (в исследовании). Выводы обоснованы, аргументированы. Оформление работы соответствует методическим рекомендациям. Объем работы соответствует требованиям. На защите обучающийся отвечает на все дополнительные вопросы;

- оценка «удовлетворительно»: обучающийся демонстрирует базовый уровень владения проблемой исследования. Во введении указаны цель и задачи исследования, но отсутствуют их четкие формулировки. Работа является компиляцией чужих исследований с попыткой формулировки собственных выводов в конце работы. Изложение материала логично и аргументировано. Наблюдается отступление от требований в оформлении и объеме работы. При ответе на вопросы обучающийся испытывает затруднения;

- оценка «неудовлетворительно»: обнаруживается несамостоятельность выполнения курсовой работы, некомпетентность в исследуемой проблеме. Нарушена логика изложения. Работа не соответствует требованиям, предъявляемым к оформлению и содержанию. На защите курсовой работы обучающийся не отвечает на вопросы.

3.3. Задания для промежуточной аттестации (экзамен, зачет)

Список вопросов к экзамену (1 семестр)

1. Каковы основные этапы разработки программного обеспечения, и почему они важны?
2. Какие свойства и способы описания алгоритма вы знаете?
3. Что такое тип данных, и какие типы данных поддерживает C++?
4. Как объявляются переменные в C++? Приведите примеры объявления переменных разных типов.
5. Что такое класс памяти, и какие классы памяти существуют в C++?
6. Что такое область действия идентификатора, и как она влияет на доступ к переменным?
7. Какие свойства переменных в C++ вам известны?
8. Какие типовые алгоритмы вы можете применить в программировании на C++? Приведите примеры.
9. Что такое синтаксические ошибки в программировании, и как их избегать?
10. Что такое семантические ошибки, и как их обнаруживать и исправлять?
11. Какие типичные алгоритмы вы можете реализовать с использованием циклов в C++?
12. Какие операторы условия (if-else) вы знаете, и как они используются для управления потоком программы?
13. Какие структуры данных поддерживает C++, и как они могут быть применены для решения задач?
14. Какие функции вы можете использовать для ввода и вывода данных в C++?
15. Предоставьте пример алгоритма на C++, который использует переменные, операторы условия и циклы для выполнения определенной задачи.
16. Что такое оператор объявления, и какие элементы он включает?
17. Какие типы данных могут быть объявлены с использованием оператора объявления в C++?
18. Какие основные операторы являются исполняемыми операторами в C++?
19. Каким образом можно встроить комментарии в ваш код C++?
20. Что такое условный оператор, и как он используется в программировании на C++?
21. Каков синтаксис условного оператора (if) в его краткой форме?
22. В чем заключается разница между краткой и полной формой условного оператора (if)?
23. Что такое оператор множественного выбора (switch), и как он применяется в C++?

24. Какие операторы могут быть использованы внутри оператора множественного выбора (switch)?
25. Что такое тернарный оператор, и как он отличается от обычного условного оператора (if)?
26. Что такое алгоритм линейной структуры, и как его можно использовать в программировании?
27. Какие операторы условного перехода вы можете использовать в C++? Приведите примеры их использования.
28. Какие операторы перехода (case) могут быть использованы внутри оператора множественного выбора (switch)?
29. Как вы можете составлять блок-схемы для алгоритмов и программ? Какие элементы должны быть включены в блок-схемы?
30. Предоставьте пример программы на C++, которая использует операторы условия (if и switch) и включает блок-схему для иллюстрации алгоритма.
31. Что такое оператор цикла с параметром for? Какой синтаксис у него?
32. Как можно использовать оператор for для вычисления суммы произведений элементов массива?
33. Что такое цикл с предусловием while? Какой синтаксис у него?
34. В чем отличие цикла с предусловием while от цикла с постусловием do-while?
35. Как работает цикл с постусловием do-while? Приведите пример использования.
36. В каких случаях следует избегать использования безусловного перехода goto в коде?
37. Какой синтаксис оператора goto и как он может быть использован в C++?
38. Какой оператор используется для прерывания выполнения цикла и перехода к следующей итерации?
39. Что делает оператор break? Как он может быть применен в циклах?
40. Какой оператор используется для перехода к следующей итерации цикла, минуя оставшуюся часть текущей итерации?
41. Какой оператор можно использовать для завершения выполнения программы внутри функции?
42. В каких случаях целесообразно использовать оператор exit для завершения программы?
43. Каким образом можно предотвратить использование бесконечных циклов?
44. Почему использование операторов goto, break, и continue может сделать код менее читаемым и поддерживаемым?
45. Предоставьте пример программы на C++, который иллюстрирует использование циклов, операторов goto, break, continue, и exit.

Список вопросов к зачету (2 семестр)

1. Что такое указатель в C++? Как объявить указатель?
2. Какие операции над указателями можно выполнять, включая арифметические операции?
3. Какие операции отношения могут быть применены к указателям? Как они работают?
4. Что такое указатель на указатель? Как его объявить и использовать?
5. Как создать массив указателей, и в каких ситуациях это может быть полезным?
6. Как строки и указатели взаимодействуют в C++?
7. Что такое ссылка в C++? Как объявить ссылку?
8. В чем отличие указателей от ссылок, и когда следует предпочесть одно другому?
9. Как можно выделить динамическую память в C++? Какие операторы и функции используются для этого?
10. Как освобождается динамически выделенная память? Какие функции используются для этой цели?
11. Что такое динамический массив? Как его создать и освободить память после использования?

12. Какие преимущества и недостатки связаны с использованием динамических массивов по сравнению с статическими массивами?
13. Как можно создать массив указателей на строки (строки) и работать с ними?
14. Какие практические сценарии требуют использования указателей и динамической памяти в программировании на C++?
15. Предоставьте пример программы на C++, который демонстрирует работу с указателями, динамической памятью и динамическими массивами.
16. Что такое прототип функции, и зачем он нужен в C++?
17. Как определяется функция в C++? Какой синтаксис используется?
18. Как осуществляется вызов функции в C++? Как передаются аргументы?
19. В чем разница между глобальными и локальными переменными? Как они влияют на область видимости?
20. Что такое классы памяти (стек и куча), и как они связаны с локальными и динамическими переменными?
21. Как передаются параметры функций в C++? Какие варианты передачи доступны?
22. Что такое значения по умолчанию для параметров функций, и как они определяются?
23. Как можно передавать массивы в функции? Какие способы передачи массивов существуют?
24. Как строки могут быть переданы в функции в C++? Какие способы передачи строк используются?
25. Что такое указатели на функции, и как они могут быть использованы?
26. Что такое рекурсивная функция, и как она отличается от обычной функции?
27. В чем состоит преимущество и недостатки использования рекурсивных функций?
28. Что такое встраиваемые функции (инлайн-функции), и как они работают?
29. Может ли функция `main()` иметь параметры? Какие параметры она может принимать, и как они используются?
30. Предоставьте пример программы на C++, который демонстрирует использование функций, передачу параметров, работу с указателями на функции и рекурсию.
31. Какие символьные функции используются для проверки, является ли символ буквой или цифрой?
32. Какие символьные функции могут быть использованы для проверки, является ли символ специальным символом?
33. Какие функции предоставляются для преобразования символов в верхний и нижний регистры?
34. Какие строковые функции позволяют проверить или обработать строки, например, для удаления пробелов?
35. Как можно преобразовать строку, представляющую число, в числовое значение?
36. Какие числовые функции предоставляются для выполнения математических операций в C++?
37. Как можно сгенерировать случайное число в C++? Какие функции используются?
38. Как представляется экран в текстовом режиме с использованием библиотеки `conio.h`?
39. Какие функции из заголовочного файла `conio.h` могут быть использованы для работы с экраном в текстовом режиме?
40. Как представляется экран в графическом режиме с использованием библиотеки `graphics.h`?
41. Какие функции из заголовочного файла `graphics.h` предоставляются для работы с экраном в графическом режиме?
42. Какие задачи чаще всего решаются с использованием символьных функций?
43. В каких сценариях полезны строковые функции для обработки и проверки строк?
44. Какие типы приложений могут использовать числовые функции и функции генерации случайных чисел?

45. В каких ситуациях экран в текстовом и графическом режимах является важным элементом разработки программы?

Список вопросов к экзамену (3 семестр)

1. Что такое пользовательский тип данных, и для чего используется ключевое слово `typedef`?
2. Как объявляется перечислимый тип данных в C++? Приведите синтаксис.
3. Какие операции можно выполнять над переменными перечислимого типа?
4. Как объявляется структура в C++? Какие данные она может содержать?
5. Что такое псевдонимы структур? Как они создаются?
6. Какие операции допустимы над переменными структурного типа?
7. Можно ли использовать массивы и структуры в качестве элементов структур?
8. Как объявляются массивы структур, и как можно обратиться к их элементам?
9. Как работают указатели на структуры в C++? Как обратиться к полям структуры с использованием указателей?
10. Какие операции можно выполнять над указателями на структуры?
11. Как объявляется объединение (`union`) в C++? В чем заключается особенность размещения объединения в памяти?
12. Какие операции допустимы с объединениями? Как они отличаются от операций со структурами?
13. Что такое битовые поля? Как они могут быть использованы?
14. Какие функции предоставляются для работы с датой и временем в C++?
15. В каких сценариях структуры, объединения и битовые поля могут быть полезными при разработке программ на C++?
16. Что такое самоссылочная структура, и в чем заключается её особенность?
17. Как классифицируются самоссылочные структуры, и какие типы самоссылочных структур вы знаете?
18. В каких областях применяются самоссылочные структуры в программировании?
19. Как можно формировать очередь с использованием самоссылочных структур?
20. Как реализовать стек с использованием самоссылочных структур?
21. Какие операции могут быть использованы для добавления и удаления элементов в односвязных списках?
22. Как формируются двусвязные списки, и какие преимущества они имеют по сравнению с односвязными списками?
23. Какие операции можно выполнять при добавлении и удалении элементов в двусвязных списках?
24. Как можно реализовать списки с помощью массивов, и в каких случаях это может быть полезно?
25. Что такое бинарное дерево, и как оно отличается от обычных деревьев?
26. В чем заключаются особенности дерева двоичного поиска? Как выполняются поиск, вставка и удаление элементов?
27. Какие основные операции могут быть выполняемы с бинарными деревьями?
28. Как можно использовать деревья для сортировки данных, и как работает сортировка на основе деревьев?
29. Какие алгоритмы сортировки вы знаете, и в чем состоят их преимущества и недостатки?
30. В каких сценариях и для каких задач наиболее эффективно использование структур данных, изученных в этих темах?
31. Какой этап включает постановку задачи и разработка функций для обслуживания базы данных?
32. Как можно сохранить базу данных на диске в C++? Какие методы используются для этого?

33. Что такое класс в C++, и как он описывается? Какие уровни доступа к элементам класса вы знаете?
34. Что такое объект в C++? Как он создается и используется?
35. Какие функции могут быть включены в состав класса, и как они вызываются?
36. Что представляет собой указатель this в классе? Как он используется?
37. Какие типы конструкторов вы можете определить для класса, и какие задачи они решают?
38. Что такое конструктор копирования, и как он отличается от других конструкторов?
39. Что выполняет деструктор в классе? Как он определяется и вызывается?
40. Как работает класс string в C++, и какие операции он поддерживает?
41. Какие конструкторы могут быть использованы для создания объектов string?
42. Как можно изменить размер строки и ее емкость в классе string?
43. Как можно сравнивать строки, и какие операции поиска поддерживаются в string?
44. Какие классификации контейнеров существуют, и в чем заключаются их различия?
45. Какие контейнеры относятся к последовательным контейнерам, и какие к ассоциативным? Какие контейнеры вы можете назвать в каждой категории?

Зачеты и экзамены могут быть проведены в письменной форме, а также в письменной форме с устным дополнением ответа. Зачеты служат формой проверки качества выполнения студентами лабораторных работ, усвоения семестрового учебного материала по дисциплине (модулю), практических и семинарских занятий (при отсутствии экзамена по дисциплине).

По итогам зачета, соответствии с модульно – рейтинговой системой университета, выставляются баллы с последующим переходом по шкале баллы – оценки за зачет, выставляемый как по наименованию «зачтено», «не зачтено», так и дифференцированно т.е. с выставлением отметки по схеме – «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно», определяемое решением Ученого совета университета и прописываемого в учебном плане.

Экзамен по дисциплине (модулю) служит для оценки работы студента в течении семестра (года, всего срока обучения и др.) и призван выявить уровень, качество и систематичность полученных им теоретических и практических знаний, приобретения навыков самостоятельной работы, развития творческого мышления, умения синтезировать полученные знания и применять их в решении практических задач. По итогам экзамена, в соответствии с модульно – рейтинговой системой университета выставляются баллы, с последующим переходом по шкале оценок на оценки: «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно», свидетельствующие о приобретенных компетенциях или их отсутствии.

Форма экзаменационного билета (пример оформления)

Министерство науки и высшего образования РФ

ФГБОУ ВО "Дагестанский государственный технический университет"

Дисциплина (Программирование)

Код, направление подготовки/специальность 090304 – Программная инженерия

Профиль (программа, специализация) Разработка программно-информационных систем

Кафедра ПОВТиАС Курс 1 Семестр 2

Форма обучения – очная/очно-заочная/заочная

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Классификация динамических структур данных
2. Очередь. Работа с очередью

Экзаменатор.....И.О.Ф.

Утвержден на заседании кафедры (протокол № от 20 г.)

Зав. кафедрой (название)Айгумов Т.Г.

Критерии оценки уровня сформированности компетенций по результатам проведения зачета:

- оценка «зачтено»: обучающийся демонстрирует всестороннее, систематическое и глубокое знание материала, свободно выполняет задания, предусмотренные программой дисциплины, усвоивший основную и дополнительную литературу. Обучающийся выполняет задания, предусмотренные программой дисциплины, на уровне не ниже базового;

- оценка «не зачтено»: обучающийся демонстрирует незнание материала, не выполняет задания, предусмотренные программой дисциплины. Обучающийся не выполняет задания, предусмотренные программой дисциплины, на уровне ниже базового. Дальнейшее освоение ОПОП не возможно без дополнительного изучения материала и подготовки к зачету.

Критерии оценки уровня сформированности компетенций по результатам проведения дифференцированного зачёта (зачета с оценкой) / экзамена:

- оценка «отлично»: обучающийся дал полный, развернутый ответ на поставленный вопрос, проявил совокупность осознанных знаний об объекте, доказательно раскрыл основные положения темы. В ответе прослеживается четкая структура,