

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ
В.Ф. УТКИНА»

Кафедра «Систем автоматизированного проектирования вычислительных средств»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

**Б1.В.ДВ.01.01 «Объектный анализ и объектно-ориентированное
программирование»**

Направление подготовки
09.04.01 Информатика и вычислительная техника

Квалификация выпускника – магистр

Форма обучения – очная, очно-заочная, заочная

Рязань 2023

1 ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов (практических заданий, описаний форм и процедур проверки), предназначенных для оценки качества освоения обучающимися данной дисциплины как части ОПОП.

Цель – оценить соответствие знаний, умений и владений, приобретенных обучающимися в процессе изучения дисциплины, целям и требованиям ОПОП в ходе проведения промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности компетенций, закрепленных за дисциплиной.

Контроль знаний обучающихся проводится в форме промежуточной аттестации. Промежуточная аттестация проводится в форме зачета.

Форма проведения зачета – письменный опрос по теоретическим вопросам.

2 ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформированность каждой компетенции (или ее части) в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

- пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;
- продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
- эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

Уровень освоения компетенций, формируемых дисциплиной

Описание критериев и шкалы оценивания теоретического вопроса:

Шкала оценивания	Критерий
3 балла (эталонный уровень)	выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя
2 балла (продвинутый уровень)	выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов
1 балл (пороговый уровень)	выставляется студенту, который дал неполный ответ на вопрос и смог ответить на дополнительные вопросы только с помощью преподавателя
0 баллов	выставляется студенту, который не смог ответить ни на один вопрос

На промежуточную аттестацию (зачет) выносятся два теоретических вопроса. Максимально студент может набрать 6 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «зачтено», «не зачтено».

Оценка «зачтено» выставляется студенту, который набрал в сумме не менее 2 баллов. Обязательным условием является выполнение всех предусмотренных в течение семестра практических и лабораторных работ заданий.

Оценка «не зачтено» выставляется студенту, который набрал в сумме менее 4 баллов, либо имеет к моменту проведения промежуточной аттестации несданные практические, либо лабораторные работы.

3. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
1	Базовые принципы ООП	ПК-1.1	Текущий контроль, Зачет
2	Объекты и классы в языке C++	ПК-1.1	Текущий контроль, Зачет
3	Наследование. Производные классы	ПК-1.1	Текущий контроль, Зачет
4	Перегрузка операторов	ПК-1.1	Текущий контроль, Зачет
5	Параметризированные классы	ПК-1.1	Текущий контроль, Зачет
6	Обработка исключений	ПК-1.1	Текущий контроль, Зачет
7	Использование библиотеки классов Qt	ПК-1.2	Текущий контроль, Зачет
8	Теоретический зачет	ПК-1.1, ПК-1.2	Зачет

4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Промежуточная аттестация в форме зачета

Код компетенции	Результаты освоения ОПОП Содержание компетенций
ПК-1	ПК-1: Способен управлять работами по сопровождению и проектами создания (модификации) информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы
ПК-1.1. Разрабатывает инструменты и методы проектирования и адаптации бизнес-процессов заказчика к возможностям информационных систем	
Знать Методы объектно-ориентированного проектирования, программирования и адаптации бизнес-процессов заказчика к возможностям информационных систем Уметь Разрабатывать объектно-ориентированные приложения для адаптации бизнес-процессов заказчика к возможностям информационных систем Владеть Объектно-ориентированным подходом для адаптации бизнес-процессов заказчика к возможностям информационных систем	
ПК-1.2. Разрабатывает организационное и технологическое обеспечение проектирования, оптимизации и дизайна информационных систем	
Знать Методы объектно-ориентированного проектирования и программирования для технологического обеспечения проектирования, оптимизации и дизайна информационных систем Уметь Разрабатывать объектно-ориентированные приложения для технологического обеспечения проектирования, оптимизации и дизайна информационных систем Владеть Объектно-ориентированной средой разработки Qt Creator для создания технологического обеспечения проектирования, оптимизации и дизайна информационных систем	

Типовые контрольные задания или иные материалы

Вопросы к зачету по дисциплине

- 1). Расскажите о базовых принципах ООП.
- 2). Перечислите основные этапы разработки программ?
- 3). Что такое класс? Чем класс отличается от объекта?
- 4). Какие правила использования базовых и производных классов?
- 5). Что такое конструктор и деструктор? Для чего они применяются?
- 6). Какие правила конструирования базовых и производных классов?
- 7). Что такое виртуальная функция? В чем ее отличие от обычной функции?
- 8). Какие правила использования виртуальных функций?
- 9). Что такое абстрактный класс? В чем его отличие от обычного класса?
- 10). Каким образом абстрактные классы используются в программах?
- 11). Что такое множественное наследование? Как оно описывается в программе?
- 12). Каким образом обеспечивается доступ к закрытым, защищенным и открытым компонентам класса в зависимости от типа наследования?
- 13). Перечислите правила использования атрибутов класса.
- 14). Для чего нужен виртуальный деструктор?
- 15). Что такое параметризованный класс (шаблон)? Для чего он используется?
- 16). Что такое параметризованные функции? Чем они отличаются от параметризованных классов?
- 17). Как переопределяются шаблоны функций?
- 18). Перечислите правила выбора одноименных функций для программы.
- 19). Как осуществляется передача дополнительных аргументов в шаблоны?
- 20). Что такое контейнеры классов и где они используются?
- 21). Как осуществляется обработка исключений в программах?
- 22). Как осуществляется определение типа исключений?
- 23). Какие основные компоненты ввода в библиотеке Qt?
- 24). Как осуществляется ввод и обработка данных приложения с помощью классов библиотеки Qt?
- 25). Каков механизм осуществления обработки событий в Qt? Чем этот механизм отличается от событийного программирования в других IDE?
- 26). Какие программируются свойства флажков и переключателей в Qt?
- 27). Как получить доступ к элементам списка библиотеки Qt?
- 28). Как получить доступ к элементам таблицы библиотеки Qt?
- 29). Как программируется меню в среде Qt Creator?
- 30). Какие функции библиотеки Qt используются для потокового ввода-вывода в приложениях?
- 31). Каким образом организуется работа с файлами с помощью библиотеки Qt?
- 32). Как можно использовать компоненты системных диалогов при разработке приложений на Qt?

Задачи для зачета по дисциплине

- 1). Постройте иерархию из двух-трех классов по заданной теме. Определите в этих классах полиморфные функции.
- 2). Постройте иерархию из двух-трех классов по заданной теме. Определите в этих классах конструкторы и деструкторы.
- 3). Опишите два наследуемых класса. Определите, какие функции этих классов могут быть виртуальными и задайте их в программе.
- 4). Задайте дружественную функцию для класса.
- 5). Опишите абстрактный класс используя чисто виртуальную функцию.
- 6). Опишите класс D, который наследовал бы свойства классов B и C, имеющих общего предка A так, чтобы класс A стал бы одним общим экземпляром для всех классов (ромбовидное наследование).
- 7). Опишите два наследуемых класса с различным доступом к компонентам класса и уровнем наследования.
- 8). Опишите виртуальный деструктор в классе.
- 9). Переопределите бинарную операцию в классе.
- 10). Переопределите унарную операцию в классе.
- 11). Переопределите бинарную операцию в функции, не являющейся членом класса.
- 12). Определите класс как шаблон. Покажите, каким образом можно использовать объекты такого класса в программе?
- 13). Определите функцию как шаблон. Покажите, каким образом можно использовать такую функцию в программе?
- 14). Переопределите заданную функцию в программе как параметризованный шаблон.
- 15). Добавьте в заданную функцию дополнительный аргумент, определяющий ограничение на размер массива.
- 16). Измените заданную программу, чтобы она работала с массивами целого и действительного типов.
- 17). Добавьте в программу обработчик исключений на неверный индекс массива.
- 18). Добавьте в программу обработчик исключений на неверный размер массива.