

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.01.02 «Объектно-ориентированное программирование»

Направление подготовки
11.03.03 Конструирование и технология электронных средств

Квалификация выпускника – бакалавр

Форма обучения – очная, заочная

Рязань 2023

Фонд оценочных средств – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных и профессиональных компетенций, приобретаемых обучающимися в соответствии с этими требованиями.

Контроль знаний обучающихся проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины (модуля), организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков обучающихся: по результатам выполнения обучающимися индивидуальных заданий в ходе практических занятий. При оценивании (определении) результатов освоения дисциплины применяется шкала оценки "зачет — незачет".

В качестве оценочных средств на протяжении семестра используются практические задания.

Результат выполнения каждого практического занятия оценивается как "зачет" в случае выполнения обучающимся всех индивидуальных заданий.

Результат выполнения каждого индивидуального задания должен соответствовать всем критериям оценки в соответствии с компетенциями, установленными для заданного раздела дисциплины.

По итогам курса обучающиеся сдают теоретический зачет и экзамен.

Форма проведения теоретического зачета – устный ответ по списку вопросов, сформулированных с учетом содержания учебной дисциплины, а также письменный ответ, содержащий решение одной практической задачи.

Форма проведения экзамена — письменный ответ по билетам.

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	ОПК-5, ПК-11	текущий контроль, выполнение ПЗ
2	Объекты и классы в языке C++	ПК-11	текущий контроль,

			выполнение ПЗ
3	Наследование. Производные классы	ПК-11	текущий контроль, выполнение ПЗ
4	Перегрузка операторов	ПК-11	текущий контроль, выполнение ПЗ
5	Параметризированные классы	ПК-11	текущий контроль, выполнение ПЗ
6	Обработка исключений	ПК-6, ПК-11	текущий контроль, выполнение ПЗ
7	Использование библиотеки классов Qt	ОПК-5, ПК-2, ПК-6, ПК-11	текущий контроль, выполнение ПЗ
8	Теоретический зачет	ОПК-5, ПК-2, ПК-6, ПК-11	Зачет

Показатели и критерии обобщенных результатов обучения

Результаты обучения по дисциплине	Показатели оценки результата	Критерии оценки результата
<p>ОПК-5 Знание принципов объектно-ориентированного программирования (ООП) для получения, хранения, переработки и трансляции информации. Умение использовать принципы ООП для получения, хранения, переработки и трансляции информации. Владение инструментальными средствами разработки программного обеспечения (ПО) на основе библиотеки Qt и языка C++ для получения, хранения, переработки и трансляции информации.</p>	<p>Выполнение проекта программы, реализующей получение, хранение, переработку и трансляцию информации.</p>	<p>Обучающийся должен продемонстрировать знание принципов ООП в ходе защиты проекта программы, реализующей получение, хранение, переработку и трансляцию информации. Обучающийся должен обеспечить соответствие структуры и содержания проекта программы, реализующей получение, хранение, переработку и трансляцию информации, принципам ООП. Обучающийся должен продемонстрировать использование инструментальных средств разработки ПО на основе библиотеки Qt и языка C++ в ходе выполнения проекта программы, реализующей получение, хранение, переработку и трансляцию информации.</p>
<p>ПК-2 Знание методов и средств разработки моделей ПО для проведения научных исследований. Умение разрабатывать модели ПО с помощью объектно-ориентированного программирования для</p>	<p>Выполнение проекта модели ПО для проведения научных исследований</p>	<p>Обучающийся должен продемонстрировать знание методов и средств разработки моделей ПО для проведения научных исследований в ходе защиты проекта программной модели. Обучающийся должен обеспечить соответствие структуры и содержания программной модели для</p>

<p>проведения научных исследований.</p> <p>Владение навыками проведения научных исследований с помощью разработанных моделей ПО на базе библиотеки Qt и языка C++.</p>		<p>проведения научных исследований объектно-ориентированному подходу.</p> <p>Обучающийся должен продемонстрировать получение результатов научных исследований с помощью разработанной программной модели на базе библиотеки Qt и языка C++.</p>
<p>ПК-6</p> <p>Знание объектно-ориентированного подхода к проектированию и верификации моделей ПО, его отличия от других подходов.</p> <p>Умение применять методы объектного анализа для верификации моделей ПО.</p> <p>Владение инструментальными средствами тестирования и верификации моделей ПО на основе библиотеки Qt и языка C++.</p>	<p>Выполнение проектирования и верификации модели ПО с использованием ООП.</p>	<p>Обучающийся должен продемонстрировать знание объектно-ориентированного подхода к проектированию и верификации моделей ПО и обосновать его отличия от других подходов.</p> <p>Обучающийся должен обеспечить реализацию методов объектного анализа при верификации модели ПО.</p> <p>Обучающийся должен продемонстрировать использование инструментальных средств тестирования и верификации модели ПО на основе библиотеки Qt и языка C++.</p>
<p>ПК-11</p> <p>Знание базовых принципов объектно-ориентированного программирования.</p> <p>Умение разрабатывать объектно-ориентированные программные средства ВТ на языке C++.</p> <p>Владение инструментальными средствами разработки программных средств ВТ на основе библиотеки Qt и языка C++.</p>	<p>Выполнение проекта программного средства ВТ</p>	<p>Обучающийся должен продемонстрировать знание базовых принципов объектно-ориентированного программирования в ходе защиты проекта программы.</p> <p>Обучающийся должен обеспечить соответствие структуры и содержания программы принципам ООП на языке C++.</p> <p>Обучающийся должен продемонстрировать использование инструментальных средств разработки программ на основе библиотеки Qt и языка C++.</p>

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 классы (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 экзамена 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.