ПРИЛОЖЕНИЕ

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.02.02 «Объектно-ориентированное программирование интеллектуальных систем»

Направление подготовки
09.03.01 Информатика и вычислительная техника

Направленность (профиль) подготовки
«Интеллектуальные системы и технологии»

Квалификация выпускника – бакалавр

Форма обучения – очная

Рязань 2026

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 обучающимся 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 выполнения практических занятий.

При оценивании (определении) результатов освоения дисциплины применяется традиционная шкала оценки (отлично, хорошо, удовлетворительно, неудовлетворительно).

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 экзамен. Форма проведения экзамена – устный ответ, по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины. В экзаменационный билет включается два теоретических вопроса по темам курса, а также одна практическая задача.

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-4.2	текущий контроль

		ПК-18.2 (PL-4.2, Б)	
2	Объекты и классы в языке C++	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
3	Наследование. Производные классы	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
4	Виртуальные функции. Абстрактные классы	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
5	Перегрузка операторов	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
6	Параметризированные функции и классы	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
7	Обработка исключений	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
8	Использование библиотеки STL при разработке интеллектуальных систем на C++	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
9	Использование библиотеки Qt при разработке интеллектуальных систем на C++	ПК-4.2 ПК-18.1 (PL-4.1, Б)	текущий контроль, выполнение ПЗ
10	Использование библиотеки TensorFlow Lite при разработке интеллектуальных систем на C++	ПК-4.2 ПК-18.1 (PL-4.1, Б) ПК-18.2 (PL-4.2, Б)	текущий контроль, выполнение ПЗ
11	Промежуточная аттестация	ПК-4.2 ПК-18.1 (PL-4.1, Б) ПК-18.2 (PL-4.2, Б)	сдача экзамена

Показатели и критерии обобщенных результатов обучения

Результаты обучения по дисциплине	Показатели оценки результата	Критерии оценки результата
ПК-4.2 **Знать** методы контроля производительности и отладки разрабатываемого программного обеспечения интеллектуальных систем **Уметь** осуществлять контроль производительности и отладки разрабатываемого программного обеспечения интеллектуальных систем с использованием Qt Creator **Владеть** средствами профилирования и отладки разрабатываемого программного обеспечения интеллектуальных систем с использованием Qt Creator	Разработка программной модели интеллектуальной системы	Обучающийся должен продемонстрировать знание методов контроля производительности и отладки разрабатываемого программного обеспечения в ходе защиты проекта программной модели. Обучающийся должен обеспечить соответствие структуры и содержания программной модели интеллектуальной системы объектно-ориентированному подходу. Обучающийся должен продемонстрировать владение средствами профилирования и отладки программной модели интеллектуальной системы на базе библиотеки Qt и языка C++.

ПК-18.1 (PL-4.1, уровень Б) **Знать** объектно-ориентированные принципы разработки приложений на C++ с использованием прикладных библиотек; методы тестирования и испытания приложений на C++ **Уметь** применять объектно-ориентированные принципы разработки приложений на C++ **Владеть** средой разработки объектно-ориентированных приложений на C++ Qt Creator	Разработка программной модели интеллектуальной системы	Обучающийся должен продемонстрировать знание базовых принципов объектно-ориентированного программирования, методов тестирования и испытания приложений на C++ в ходе защиты проекта программы. Обучающийся должен обеспечить соответствие структуры и содержания разрабатываемого приложения принципам ООП на C++. Обучающийся должен продемонстрировать владение инструментальными средствами разработки объектно-ориентированных приложений на C++ Qt Creator.
ПК-18.2 (PL-4.2, уровень Б) **Знать** объектно-ориентированный подход разработки приложений на C++ с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем **Уметь** применять объектно-ориентированные принципы разработки и отладки приложений на C++ с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем **Владеть** средой разработки объектно-ориентированных приложений на C++ Qt Creator с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем	Разработка программной модели интеллектуальной системы	Обучающийся должен продемонстрировать знание объектно-ориентированного подхода разработки приложений на C++ с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем, в ходе защиты проекта программы. Обучающийся должен обеспечить соответствие структуры и содержания разрабатываемого приложения на C++ принципам ООП с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем. Обучающийся должен продемонстрировать владение инструментальными средствами разработки объектно-ориентированных приложений на C++ Qt Creator с использованием библиотек TFLite и Qt под конкретные аппаратные платформы, в том числе для встроенных систем

Типовые контрольные задания или иные материалы

Вопросы для подготовки к экзамену 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 витруальная 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 параметризированный классы (шаблон)? Для чего он используется?
16). Что такое параметризированные функции? Чем они отличаются от параметризированных 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). Какие бывают итераторы в шаблонах классов STL и чем они отличаются?
22). Как осуществляется обработка исключений в программах?
23). Как осуществляется определение типа исключений?
24). Какие основные компоненты ввода в библиотеке Qt?
25). Как осуществляется ввод и обработка данных приложения с помощью классов библиотеки Qt?
26). Каков механизм осуществления обработки событий в Qt? Чем этот механизм отличается от событийного программирования в других IDE?
27). Какие программируются свойства флажков и переключателей в Qt?
28). Как получить доступ к элементам списка библиотеки Qt?
29). Как получить доступ к элементам таблицы библиотеки Qt?
30). Как программируется меню в среде Qt Creator?
31). Какие функции библиотеки Qt используются для потокового ввода-вывода в приложениях?
32). Каким образом организуется работа с файлами с помощью библиотеки Qt?
33). Какие компоненты системных диалогов можно использовать при разработке приложений на Qt?
34). Каким образом можно интегрировать библиотеку TensorFlow Lite в приложение на Qt?
35). Какие функции библиотеки TensorFlow Lite можно использовать в приложениях 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 параметризированный 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.

Оператор ЭДО ООО "Компания "Тензор"

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

ПОДПИСАНО ЗАВЕДУЮЩИМ КАФЕДРЫ	**ФГБОУ ВО "РГРТУ", РГРТУ,** Корячко Вячеслав Петрович, Заведующий кафедрой САПР	**20.10.26** 15:14 (MSK)	Простая подпись
ПОДПИСАНО ЗАВЕДУЮЩИМ ВЫПУСКАЮЩЕЙ КАФЕДРЫ	**ФГБОУ ВО "РГРТУ", РГРТУ,** Корячко Вячеслав Петрович, Заведующий кафедрой САПР	**20.10.26** 15:14 (MSK)	Простая подпись