

Минобрнауки России
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ



Заведующий кафедрой
Сирота Александр Анатольевич

Кафедра технологий обработки и защиты информации

23.04.2024

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.ДВ.05.01 Разработка приложений на C++

1. Код и наименование направления подготовки/специальности:

10.05.01 Компьютерная безопасность

2. Профиль подготовки/специализация:

Анализ безопасности компьютерных систем

3. Квалификация (степень) выпускника:

Специалитет

4. Форма обучения:

Очная

5. Кафедра, отвечающая за реализацию дисциплины:

Кафедра технологий обработки и защиты информации

6. Составители программы:

Дрюченко Михаил Анатольевич, к.т.н., доцент

7. Рекомендована:

протокол №5 от 05.03.2024

8. Учебный год:

2026-2027

9. Цели и задачи учебной дисциплины:

приобретение базовых знаний и навыков по алгоритмизации, разработке, отладке и тестированию программ на языке C++, проектированию и разработке приложений с применением объектно-ориентированного подхода.

Основные задачи дисциплины:

изучение технологии программирования на языке C++;

раскрытие принципов структурного, модульного, объектно-ориентированного подхода при проектировании и разработке приложений;

овладение средствами объектно-ориентированного программирования языка C++,

средствами стандартной библиотеки STL.

изучение методов отладки и тестирования программ на C++.

10. Место учебной дисциплины в структуре ООП:

Блок обязательные дисциплины вариативной части.

Для успешного освоения дисциплины необходимы входные знания в области алгебры и геометрии,

математической логики и теории алгоритмов, информатики.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников) и индикаторами их достижения:

| Код и название компетенции | Код и название индикатора компетенции | Знания, умения, навыки |
|---|---|---|
| ПК-1 Способен проводить анализ требований к программному обеспечению, выполнять работы по проектированию программного обеспечения | ПК-1.1 Знает методологии и технологии разработки программного обеспечения и технологии программирования | Знает методологии и технологии разработки программного обеспечения и технологии программирования на языке C++. Умеет использовать стандартные инструменты и среды программирования для разработки, отладки и тестирования программ на языке C++. Владеет практическими навыками использования современных программных средств для разработки, отладки и тестирования приложений на языке C++. |
| ПК-1 Способен проводить анализ требований к программному обеспечению, выполнять работы по проектированию программного обеспечения | ПК-1.2 Знает применяемые математические методы и алгоритмы функционирования для компонентов программных средств | Знает алгоритмы функционирования компонентов программных средств, используемых для разработки, сборки и отладки программ на языке C++. Умеет использовать стандартные инструменты и среды программирования для разработки и отладки программ на языке C++. Владеет практическими навыками использования современных программных средств для разработки, отладки и тестирования приложений на языке C++. |

| Код и название компетенции | Код и название индикатора компетенции | Знания, умения, навыки |
|---|---|---|
| ПК-1 Способен проводить анализ требований к программному обеспечению, выполнять работы по проектированию программного обеспечения | ПК-1.3 применять технологии обработки данных и анализировать возможности их использования при разработке программного обеспечения в профессиональной деятельности | Знает основные алгоритмы и структуры данных, используемые при разработке программного обеспечения. Умеет разрабатывать программы, реализующие различные структуры и алгоритмы обработки данных. Владеет практическими навыками использования инструментов и сред программирования для разработки и отладки программ на языке C++. |

12. Объем дисциплины в зачетных единицах/час:

2,5/90

Форма промежуточной аттестации:

Зачет с оценкой

13. Трудоемкость по видам учебной работы

| Вид учебной работы | Семестр 5 | Семестр 6 | Всего |
|--------------------------|-----------|-----------|-------|
| Аудиторные занятия | 72 | | 72 |
| Лекционные занятия | 36 | | 36 |
| Практические занятия | 0 | | 0 |
| Лабораторные занятия | 36 | | 36 |
| Самостоятельная работа | 18 | | 18 |
| Курсовая работа | 0 | | 0 |
| Промежуточная аттестация | 0 | | 0 |
| Часы на контроль | 0 | | 0 |
| Всего | 90 | | 90 |

13.1. Содержание дисциплины

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---------------------------------|-------------------------------|--|
| 1. | Лекции | | |

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---|--|--|
| 1.1 | Начальные сведения о языке программирования С++ | <p>Структура программного обеспечения и этапы его разработки.</p> <p>Технология разработки программ на языке С++ (создание исходного файла, компиляция, компоновка, отладка, тестирование, сопровождение).</p> <p>Конструкции и объекты языка С++ (операции, управляющие последовательности, строковые литералы, идентификаторы, константы, типы, классы памяти и т.д.).</p> <p>Указатели и операции над ними.</p> <p>Динамическое распределение памяти.</p> <p>Директивы препроцессора.</p> <p>Структурное программирование.</p> <p>Модульное программирование.</p> <p>Встроенные и библиотечные функции.</p> | <p>Создан онлан электронный курс, размещены материалы к лекции.</p> <p>Размещены индивидуальные задания для выполнения лабораторных работ.</p> |

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|--|---|--|--|
| 1.2 | Основы объектно-ориентированного программирования | <p>Элементы объектно-ориентированного программирования. Инкапсуляция. Наследование. Полиморфизм. Классы и объекты. Конструкторы и деструкторы. Дружественные функции. Перегрузка операций. Доступ к базовым и производным классам. Виртуальные методы. Простое и множественное наследование. Шаблоны классов.</p> | <p>Создан онлан электронный курс, размещены материалы к лекции. Размещены индивидуальные задания для выполнения лабораторных работ.</p> |
| 1.3 | Разработка приложений с графическим интерфейсом | Изучение возможностей интегрированной среды Microsoft Visual C++/Qt Creator для разработки приложений с графическим интерфейсом | <p>Создан онлан электронный курс, размещены материалы к лекции. Размещены индивидуальные задания для выполнения лабораторных работ.</p> |
| 2. Практические занятия | | | |
| 2.1 | нет | | |
| 3. Лабораторные работы | | | |

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---|---|---|
| 3.1 | Начальные сведения о языке программирования С++ | <p>1. Изучение и использование интегрированной среды разработки программ Microsoft Visual Studio. Исследование базовых возможностей среды разработки Microsoft Visual Studio. Приобретение практических навыков по созданию на ее основе простейших программ на языке С++, реализующих простейшие операции, а также ввод и вывод информации.</p> <p>2. Изучение стандартных типов данных С++, переменных, операций и выражений.</p> <p>3. Изучение особенностей использования препроцессора, а также вспомогательных математических функций, функций локализации, функций работы с датой и временем, функций классификации и преобразования символов, различных функций в стиле языка С (ввода-вывода и пр.).</p> <p>4. Изучение операторов выбора, операторов циклов, операторов передачи управления, а также принципов работы с одномерными и многомерными массивами.</p> <p>5. Изучение механизмов работы со структурами данных, механизмов объявления, определения, передачи параметров при создании собственных функций.</p> | Размещены индивидуальные задания для выполнения лабораторных работ. |

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---|--|---|
| 3.2 | Основы объектно-ориентированного программирования | 6. Практическое изучение основных концепций объектно-ориентированного программирования (классов и объектов). Программирование классов. 7. Изучение шаблонов и дружественных функций. | Размещены индивидуальные задания для выполнения лабораторных работ. |
| 3.3 | Разработка приложений с графическим интерфейсом | 8. Изучение возможностей интегрированной среды Microsoft Visual C++/Qt Creator для разработки приложений с графическим интерфейсом | Размещены индивидуальные задания для выполнения лабораторных работ. |

13.2. Темы (разделы) дисциплины и виды занятий

| № п/п | Наименование темы (раздела) | Лекционные занятия | Практические занятия | Лабораторные занятия | Самостоятельная работа | Всего |
|-------|---|--------------------|----------------------|----------------------|------------------------|-------|
| 1 | Начальные сведения о языке программирования C++ | 12 | | 12 | 4 | 32 |
| 2 | Основы объектно-ориентированного программирования | 16 | | 16 | 8 | 48 |
| 3 | Разработка приложений с графическим интерфейсом | 8 | | 8 | 6 | 28 |
| | | 36 | 0 | 36 | 18 | 72 |

14. Методические указания для обучающихся по освоению дисциплины

1) При изучении дисциплины рекомендуется использовать следующие средства:
рекомендуемую основную и дополнительную литературу;
методические указания и пособия;
контрольные задания для закрепления теоретического материала;

электронные версии учебников и методических указаний для выполнения лабораторно - практических работ (при необходимости материалы рассылаются по электронной почте).

2) Для максимального усвоения дисциплины рекомендуется проведение письменного опроса (тестирование, решение задач) студентов по материалам лекций и практических работ. Подборка вопросов для тестирования осуществляется на основе изученного теоретического материала. Такой подход позволяет повысить мотивацию студентов при конспектировании лекционного материала.

3) При проведении лабораторных занятий обеспечивается максимальная степень соответствия с материалом лекционных занятий и осуществляется экспериментальная проверка материалов излагаемых в рамках лекций.

4) При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей, вовремя подключаться к online занятиям, ответственно подходить к заданиям для самостоятельной работы

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

| № п/п | Источник |
|-------|--|
| 1 | Разработка приложений на С++ : учебно-методическое пособие / М.А. Дрюченко, Е.Ю. Митрофанова ; Воронеж. гос. ун-т .— Воронеж : Издательский дом ВГУ, 2019 .— 114 с. — Библиогр.: с. 114. |

б) дополнительная литература:

| № п/п | Источник |
|-------|---|
| 1 | Павловская, Татьяна Александровна. С++. Объектно-ориентированное программирование : практикум : учебное пособие для студ. вузов, обуч. по направлению подгот. дипломир. специалистов "Информатика и вычислительная техника" / Т.А. Павловская, Ю.А. Щупак .— СПб. [и др.] : Питер, 2008 .— 264 с. : ил .— (Учебное пособие) .— Библиогр.: с.260 .— Алф. указ.: с.261-264 .— ISBN 978-5-94723-842-6. |
| 2 | Павловская, Татьяна Александровна. С/С++. Программирование на языке высокого уровня : учебник для студ. вузов, обуч. по направлению "Информатика и вычислительная техника" / Т.А. Павловская .— СПб. [и др.] : Питер, 2008 .— 460 с. : ил .— (Учебник для вузов) .— Предм. указ.: с.450-460 .— ISBN 978-5-94723-568-5. |
| 3 | Прата, Стивен. Язык программирования С++ : лекции и упражнения / Стивен Прата ; пер. с англ. Д.Я. Иваненко [и др.]; под ред. Ю.Н. Артеменко .— 5-е изд. — М. ; СПб ; Киев : Вильямс, 2007 .— 1181 с. : ил .— Парал. тит. л. англ. — Предм. указ.: с.1175-1181 .— ISBN 5-8459-1127-3. |
| 4 | Шилдт, Герберт. С++ : базовый курс / Герберт Шилдт ; [пер. с англ. и ред. Н.М. Ручко] .— 3-е изд. — М.[и др.] : Вильямс, 2011 .— 620 с. : ил. — Парал. тит. л. англ. — Предм. указ.: с.610-620 .— ISBN 978-5-8459-0768-4. |

в) информационные электронно-образовательные ресурсы:

| № п/п | Источник |
|-------|---|
| 1 | Электронный каталог Научной библиотеки Воронежского государственного университета. – (http // www.lib.vsu.ru/). |
| 2 | Образовательный портал «Электронный университет ВГУ».– (https://edu.vsu.ru/) |
| 3 | ЭБС Лань, Лицензионный договор №3010, (с 01/03/2024 по 28.02.2025) 06/02 24 от 13.02.2024 (с дополнительным соглашением №1 от 14.03.2024) |
| 4 | ЭБС «Университетская библиотека online» (Контракт №3010 06/11 23 от 26.12.2023 (с 26.12.2023 по 25.12.2024) |
| 5 | ЭБС «Консультант студента» – Лицензионный договор №980КС/12-2023 / 3010-06/01-24 от 24.01.2024 с 24.01.2024 по 11. 01.2025) |
| 6 | Электронная библиотека ВГУ, Договор №ДС-208 от 01.02.2021 с ООО «ЦКБ «БИБКОМ» и ООО «Агентство «Книга-Сервис» о создании Электронной библиотеки ВГУ, (с 01.02.2021 по 31.01.2027) |
| 7 | БС ВООК.ru, Договор №3010 15/983 23 от 20.12.2023, (с 01.02.2024 по 31.01.2025) |

16. Перечень учебно-методического обеспечения для самостоятельной работы

| № п/п | Источник |
|-------|---|
| 1 | Павловская, Татьяна Александровна. С++. Объектно-ориентированное программирование : практикум : учебное пособие для студ. вузов, обуч. по направлению подгот. дипломир. специалистов "Информатика и вычислительная техника" / Т.А. Павловская, Ю.А. Щупак . – СПб. [и др.] : Питер, 2008 .— 264 с. : ил .— (Учебное пособие) .— Библиогр.: с.260 .— Алф. указ.: с.261-264 .— ISBN 978-5-94723-842-6. |
| 2 | Основы программирования на С++ : учебно-методическое пособие для вузов / Воронеж. гос. ун-т ; сост. Ю.А. Крыжановская .— Воронеж : ИПЦ ВГУ, 2008 .— 23 с. — Библиогр.: с.23 .— <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/m08-32.pdf >. |

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

Для реализации учебного процесса используется:

ПО ОС Windows v.7, 8, 10, Visual Studio, v. 2010-2019.

При проведении занятий в дистанционном режиме обучения используются технические и информационные ресурсы Образовательного портала "Электронный университет ВГУ"

(<https://edu.vsu.ru>), базирующегося на системе дистанционного обучения Moodle, развернутой в университете.

18. Материально-техническое обеспечение дисциплины:

1) 394018, г. Воронеж, площадь Университетская, д. 1, корпус 1а, ауд. 479

Учебная аудитория: специализированная мебель, компьютер преподавателя i5-8400-2,8ГГц, монитор с ЖК 19", мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Visual Studio, v. 2010-2019, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Foxit PDF Reader.

2) 394018, г. Воронеж, площадь Университетская, д. 1, корпус 1а, аудитория 292

Учебная аудитория: специализированная мебель, компьютер преподавателя Pentium-G3420-3,2ГГц, монитор с ЖК 17", мультимедийный проектор, экран. Система для видеоконференций Logitech ConferenceCam Group и ноутбук 15.6" FHD Lenovo V155-15API

ПО: ОС Windows v.7, 8, 10, Visual Studio, v. 2010-2019, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Foxit PDF Reader.

3) 394018, г. Воронеж, площадь Университетская, д. 1, корпус 1а, ауд. 380

Учебная аудитория: специализированная мебель, компьютер преподавателя i3-3240-3,4ГГц, монитор с ЖК 17", мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Visual Studio, v. 2010-2019, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Foxit PDF Reader.

4) 394018, г. Воронеж, площадь Университетская, д. 1, корпус 1а, аудитория 290

Учебная аудитория: специализированная мебель, персональные компьютеры на базе i7-7800x-4ГГц, мониторы ЖК 27" (12 шт.), мультимедийный проектор, экран.

Лабораторное оборудование искусственного интеллекта: рабочие места – персональные компьютеры на базе i7-7800x-4ГГц, мониторы ЖК 27" (12 шт.); модули АО НПЦ «ЭЛВИС»: процессорный Салют-ЭЛ24ПМ2 (9 шт.), отладочный Салют-ЭЛ24ОМ1 (9 шт.), эмулятор MC-USB-JTAG (9 шт.).

Лабораторное оборудование электроники, электротехники и схемотехники: рабочие места – персональные компьютеры на базе i7-7800x-4ГГц, мониторы ЖК 27" (12 шт.); стенд для практических занятий по электрическим цепям (KL-100); стенд для изучения аналоговых электрических схем (KL-200); стенд для изучения цифровых схем (KL-300).

ПО: ОС Windows v.7, 8, 10, Visual Studio, v. 2010-2019, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Foxit PDF Reader

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

| № п/п | Разделы дисциплины (модули) | Код компетенции | Код индикатора | Оценочные средства для текущей аттестации |
|-------|---|-----------------|----------------|---|
| 1 | Разделы 1-3 Начальные сведения о языке программирования C++. Основы объектно-ориентированного программирования. Разработка приложений с графическим интерфейсом | ПК-1 | ПК-1.1 | Устный опрос, Лабораторные работы 1-8 |

| № п/п | Разделы дисциплины (модули) | Код компетенции | Код индикатора | Оценочные средства для текущей аттестации |
|-------|---|-----------------|----------------|---|
| 2 | Разделы 1-3 Начальные сведения о языке программирования С++. Основы объектно-ориентированного программирования. Разработка приложений с графическим интерфейсом | ПК-1 | ПК-1.2 | Устный опрос, Лабораторные работы 1-8 |
| 3 | Разделы 1-3 Начальные сведения о языке программирования С++. Основы объектно-ориентированного программирования. | ПК-1 | ПК-1.3 | Устный опрос, Лабораторные работы 1-8 |

Промежуточная аттестация

Форма контроля – зачет с оценкой

Оценочные средства для промежуточной аттестации

Перечень вопросов, лабораторные работы

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Текущий контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Устный опрос на практических занятиях

Контрольная работа по теоретической части курса

Лабораторные работы

Примерный перечень применяемых оценочных средств

| № п/п | Наименование оценочного средства | Представление оценочного средства в фонде | Критерии оценки |
|-------|----------------------------------|---|--|
| 1 | 2 | 3 | 4 |
| 1 | Устный опрос | Вопросы по темам/разделам дисциплины | Правильный ответ – зачтено, неправильный или принципиально неточный ответ - не зачтено |

| | | | |
|---|---|--|---|
| 2 | Контрольная работа по разделам дисциплины | Теоретические вопросы по темам/разделам дисциплины | Шкала оценивания соответствует таблице, приведенной ниже |
| 3 | Лабораторная работа | Содержит 8 лабораторных заданий, предусматривающих разработку, отладку и тестирование программ на языке C++ | При успешном выполнении работы ставится оценка зачтено и осуществляется допуск к экзамену, в противном случае ставится оценка не зачтено и обучающийся не допускается к экзамену. |
| 4 | КИМ промежуточной аттестации | Каждый контрольно-измерительный материал для проведения промежуточной аттестации включает 2 задания вопросов для контроля знаний, умений и владений в рамках оценки уровня сформированности компетенции. | Шкала оценивания приведена ниже |

Пример задания для выполнения лабораторной работы

Лабораторная работа № 2

«Изучение типов данных, операций и выражений»

Цель работы

Изучение стандартных типов данных C++, переменных, операций и выражений.

Форма контроля

Опрос в устной форме по исходному коду и результатам работы реализованной программы

Количество отведённых аудиторных часов - 4

Содержание работы

Получить у преподавателя вариант задания, написать код, реализующий соответствующий алгоритм обработки информации. Отладить и протестировать программу.

Пример вариантов заданий:

Составить программу «калькулятор», выполняющую стандартные арифметические операции с данными целых и вещественных типов.

Примеры контрольных вопросов:

1. Что такое выражение? Из чего состоит выражение?
2. Перечислить все типы данных, которые существуют в C++. Сколько места в памяти занимают данные каждого типа?
3. Какие операции можно применять к целочисленным данным? К вещественным данным? К символьным данным?
4. Привести пример тернарной операции.

Приведённые ниже задания рекомендуется использовать при проведении диагностических работ для оценки остаточных знаний по дисциплине.

Компетенция ПК-1.1

Задания закрытого типа

1. Выражения могут содержать

- а) только бинарные операции;
- б) только тернарные операции;
- в) только унарные или бинарные;
- г) унарные, бинарные или тернарные

2. Чему будет равно x

```
int x=1;
x = ++x + ++x;
cout << x << endl;
```

- а) 5
- б) 6
- в) 4

3. Чему будет равно x

```
int x=1;
x = x++ + ++x;
cout << x << endl;
```

- а) 5
- б) 6
- в) 4

4. Выберите логические операции (возможен множественный выбор)

- а) ~
- б) ||
- в) |
- г) &&
- д) ==

5. Выберите побитовые операции (возможен множественный выбор)

- а) +
- б) &
- в) ||
- г) !=
- д) <<

6. Чему будет равно z

```
uint8_t x=1, y=0, z=!((x&& y) || y);
```

- а) 1
- б) 0
- в) 255

7. Чему будет равно z

```
uint8_t x=1, y=0, z=~((x&& y) || y);
```

- а) 1
- б) 0
- в) 255

8. Что будет выведено в результате выполнения фрагмента кода

```
int i=4, j=0;
do j+=+i; while(i<-3);
cout << i << ' ' << j;
```

- a) 5 5
- б) 4 0
- в) 5 4

9. Что будет выведено в результате выполнения фрагмента кода

```
int a[]={2,2,2,5,5,3,5,4,2,1}, *p=a;
while((p+=*p) < a+10) cout << ' ' << *p;
```

- a) 2 2 4
- б) 2 5 1
- в) 2 5 2

10. При вычислении выражений последовательность выполнения операций определяется

- a) только скобками;
- б) только приоритетом операций;
- в) скобками, приоритетом операций, а при одинаковом приоритете ассоциативностью операций

Задания открытого типа

1. Распишите вывод для фрагмента кода

```
char *s="hnmmasoazermjrgtrzd",
*p[]={s+3, s+3, s+4, s+4, s+2}, **pp=p;
cout << s[2] << *p[3] << p[1][2] << pp[2][4];
```

2. Распишите вывод для фрагмента кода

```
int i=0, j=-1, k=-1;
if(++i || ++j && ++k) cout << i << ' ' << j << ' ' << k;
else cout << -i << ' ' << -j << ' ' << -k;
```

3. Распишите вывод для фрагмента кода

```
int i=-9, *p=&i;
cout << ++*p; cout << ' ' << *p++;
cout << ' ' << ++*--p << ' ' << i ;
```

4. Распишите вывод для фрагмента кода

```
char *s="ldjxjpxovjtoalmi";
int i, a[]={6,0,4,2,3,7,0,4,3,8};
for(i=0; i<10; i++) cout << *(s+(a+i));
```

Задания с развёрнутым ответом

1. Дайте краткое описание шаблонов функций и классов. Приведите определение специализации, конкретизации, инстанцирования шаблонов. Приведите примеры кода с описанием шаблонных функций и классов.

| Критерии оценивания | Шкала оценок |
|---|--------------|
| Обучающийся приводит развернутое и безошибочное описание шаблонов функций и классов. Приводит корректные определения специализации, конкретизации, инстанцирования шаблонов и примеры кода с описанием шаблонных функций и классов. | 3 балла |
| Обучающийся приводит достаточно развернутое описание шаблонов функций и классов, определений специализации, конкретизации, инстанцирования шаблонов. Приводит примеры кода с описанием шаблонных функций и классов. В описании допускаются незначительные неточности. | 2 балла |
| Представлено недостаточно развернутое описание шаблонов функций и классов, в котором могут содержаться отдельные неточности. Приведенные примеры кода шаблонных функций и классов, содержат ошибки. | 1 балл |
| Представлено неполное или содержащее грубые ошибки описание шаблонов функций и классов. Примеры кода отсутствуют. | 0 баллов |

2. Опишите механизмы наследования в C++. Приведите примеры кода, демонстрирующие общее, частное, множественное наследование.

| Критерии оценивания | Шкала оценок |
|---|--------------|
| Обучающийся приводит развернутое и безошибочное описание механизмов наследования в C++. Приводит корректные примеры кода, демонстрирующие общее, частное, множественное наследование. | 3 балла |
| Обучающийся приводит достаточно развернутое описание механизмов наследования в C++. Приводит корректные примеры кода, демонстрирующие общее, частное, множественное наследование. В описании допускаются незначительные неточности. | 2 балла |
| Представлено недостаточно развернутое описание механизмов наследования в C++, в котором могут содержаться отдельные неточности. Приведенные примеры кода демонстрирующие общее, частное, множественное наследование, содержат ошибки. | 1 балл |
| Представлено неполное или содержащее грубые ошибки описание механизмов наследования в C++. Примеры кода отсутствуют. | 0 баллов |

Компетенция ПК-1.2

Задания закрытого типа

1. Выберите каким определениям функции может соответствовать вызов

`f(3.14)`

- а) `void f(double x);`
- б) `void f(float x);`
- в) `void f(float x, float y);`
- г) `void f(float x, float y=0);`
- д) `float f(float x, float y=0);`
- е) `void f(float x=0, float y);`

2. Выберите корректный вариант освобождения памяти для динамического массива

`int *buf= new char[20];`

- a) free(buf);
- б) delete[] buf;
- в) delete buf;
- г) delete buf[];

3. Разница между формальными и фактическими параметрами функции заключается в том, что

- а) первые определены в заголовке функции, а вторые – представляют собой значения, с которыми функция вызывается;
- б) первые могут передаваться по значению, а вторые – по ссылке;
- в) первые могут быть константными, а вторые – нет;

4. Выберите корректный вариант описания шаблонной функции (возможен множественный выбор)

- a) `template <typename T> void f(T x, T y) { ... };`
- б) `template <class T>; void f<T x, T y> { ... };`
- в) `template <class T> void f(T& x, T& y) { ... };`
- г) `template class T void f(T x, T x) (...);`

5. Число параметров шаблона при определении шаблона функции определяется

- а) числом параметризуемых типов;
- б) числом аргументов функции

6. Выберите корректный вариант создания экземпляра объекта шаблонного класса

```
template <class T> class Vector
{
    public:
        Vector(int) { ... }
        ~ Vector () { ... }
}
```

- а) `Vector v(10);`
- б) `Vector::<int> v(10);`
- в) `Vector<int> v(10);`

7. Что будет выведено в результате выполнения фрагмента кода

```
class A
{ public: void f() { cout << "A"; } };
class B : public A
{ public: void f() { cout << "B"; } };
B b;
A&a=b;
a.f();
```

- а) BA
- б) B
- в) A
- г) AB

8. Что обозначает объявление

```
extern int x;
```

- а) переменная x не может быть изменена вне текущего модуля;
- б) переменная x определена в другом исходном файле;
- в) переменная x является статической

9. Этапы компиляции программы

- а) препроцессинг → ассемблирование → компиляция → линковка
- б) компиляция → препроцессинг → ассемблирование → линковка
- в) компиляция → линковка → препроцессинг → ассемблирование
- г) препроцессинг → компиляция → ассемблирование → линковка

Задания открытого типа

- 1. Структура данных, представленная в виде группы ячеек одного типа, объединенных под единым именем, применяемая для обработки большого количества однотипных данных
- 2. Особый тип переменной, хранящей адрес некоторой другой переменной
- 3. Один из способов реализации полиморфизма – возможность одновременного существования в одной области видимости нескольких различных вариантов применения оператора, имеющих одно и то же имя, но различающихся типами параметров, к которым они применяются

Задания с развёрнутым ответом

1. Опишите средства обработки ошибок и исключений в C++. Напишите пример кода, демонстрирующий использование инструкций try, catch, throw.

| Критерии оценивания | Шкала оценок |
|---|--------------|
| Обучающийся приводит развернутое и безошибочное описание средств обработки ошибок и исключений в C++. Приводит корректные примеры, демонстрирующие использование инструкций try, catch, throw. | 3 балла |
| Обучающийся приводит достаточно развернутое описание средств обработки ошибок и исключений в C++. В описании допускаются незначительные неточности. Приводит корректные примеры, демонстрирующие использование инструкций try, catch, throw. | 2 балла |
| Представлено недостаточно развернутое описание средств обработки ошибок и исключений в C++, в котором могут содержаться отдельные неточности. Приведенные примеры кода, демонстрирующие использование инструкций try, catch, throw, также могут содержать неточности. | 1 балл |
| Представлено неполное или содержащее грубые ошибки описание средств обработки ошибок и исключений в C++. Примеры кода отсутствуют. | 0 баллов |

2. Опишите правила преобразования типов в C++ (static_cast, const_cast, dynamic_cast, reinterpret_cast). Приведите примеры кода, демонстрирующие применение операторов преобразования типов.

| Критерии оценивания | Шкала оценок |
|---|--------------|
| Обучающийся приводит развернутое и безошибочное описание правил преобразования типов в C++. Приводит корректные примеры кода, демонстрирующие применение операторов преобразования типов. | 3 балла |
| Обучающийся приводит достаточно развернутое описание правил преобразования типов в C++, а также примеры кода. В описании допускаются незначительные неточности. | 2 балла |
| Представлено частичное или недостаточно развернутое описание правил преобразования типов в C++, в котором могут содержаться отдельные неточности. Приведенные примеры кода содержат ошибки. | 1 балл |
| Представлено неполное или содержащее грубые ошибки описание правил преобразования типов в C++. Примеры кода отсутствуют. | 0 баллов |

Компетенция ПК-1.3

Задания закрытого типа

1. Классы необходимы для

- а) обеспечения возможности взаимодействия кода из разных модулей приложения;
- б) определения пользовательских типов данных;
- в) объединения данных и операций над ними;

2. По умолчанию поля класса объявляются как

- а) private
- б) public
- в) protected

3. Определение класса – это

- а) вызов конструктора;
- б) объявление всех его методов и полей;
- в) инициализация всех его полей

4. Какая называется функция, не являющаяся методом класса, но имеющая доступ к его защищенным и внутренним членам

- а) шаблонная;
- б) статическая;
- в) дружественная

5. Виртуальные функции – это

- а) функции базового класса, которые не используются в производном классе;
- б) функции базового класса, которые могут быть переопределены в производном классе
- в) функции базового класса, которые не имеют реализации

6. Дружественная функция – это

- а) функция другого класса, среди аргументов которой есть элементы данного класса
- б) функция, являющаяся членом класса и объявленная с атрибутом friend;
- в) функция, объявленная в классе с атрибутом friend, но не являющаяся членом класса

7. Выберите утверждения, справедливые для абстрактных классов (возможен множественный выбор)

- а) объект такого класса создать невозможно;
- б) в таком классе нельзя определять методы;
- в) класс, имеющий хотя бы один чисто виртуальный метод;
- г) такой класс не может иметь наследников

8. Механизмы, через которые реализуется полиморфизм

- а) перегрузка функций, наследование, шаблоны;
- б) перегрузка функций, виртуальные функции, шаблоны;
- в) наследование, виртуальные функции, дружественные функции

9. Можно ли в шаблоне класса определить статический метод?

- а) да;
- б) нет

10. Статические атрибуты класса (возможен множественный выбор)

- а) существуют в единственном экземпляре, независимо от количества объектов;
- б) не могут изменяться;
- в) инициализируются при создании первого объекта;
- г) инициализируются в начале выполнения программы

11. Выберите утверждения, которые можно считать справедливыми при использовании шаблонов (возможен множественный выбор)

- а) повторное использование кода;
- б) уменьшение времени компиляции;
- в) возможность использования шаблонных реализаций интерфейсов при построении иерархий наследования;
- г) упрощение анализа кода, содержащего шаблоны

Задания открытого типа

1. Функции, которые не являются членами класса, однако имеют доступ к его закрытым членам – полям и методам в секции private
2. Событие при выполнении программы, которое приводит к ее ненормальному или неправильному поведению (бывают аппаратные и программные)

Задания с развёрнутым ответом

1. Опишите последовательные и ассоциативные контейнеры стандартной библиотеки C++. Приведите практические рекомендации по использованию каждого типа контейнера. Приведите примеры кода, демонстрирующие использование некоторых контейнеров.

| Критерии оценивания | Шкала оценок |
|---|--------------|
| Обучающийся приводит развернутое и безошибочное описание контейнеров стандартной библиотеки C++. Приводит корректные практические рекомендации по использованию каждого типа контейнера, а также примеры кода. | 3 балла |
| Обучающийся приводит достаточно развернутое описание контейнеров стандартной библиотеки C++. Приводит практические рекомендации по использованию каждого типа контейнера, а также примеры кода. В описании допускаются незначительные неточности. | 2 балла |
| Представлено недостаточно развернутое описание контейнеров стандартной библиотеки C++, в котором могут содержаться отдельные неточности. Приведенные примеры кода и рекомендации по использованию каждого типа контейнера содержат ошибки. | 1 балл |
| Представлено неполное или содержащее грубые ошибки описание контейнеров стандартной библиотеки C++. Примеры кода и рекомендации по использованию каждого типа контейнера отсутствуют. | 0 баллов |

20.2 Промежуточная аттестация

Промежуточная аттестация может включать в себя проверку теоретических вопросов, а также, при необходимости (в случае невыполнения в течение семестра), проверку выполнения установленного перечня лабораторных заданий, позволяющих оценить уровень полученных знаний и/или практическое (ие) задание(я), позволяющее (ие) оценить степень сформированности умений и навыков.

Для оценки теоретических знаний используется перечень контрольно-измерительных материалов. Каждый контрольно-измерительный материал для проведения промежуточной аттестации включает два задания - вопросов для контроля знаний, умений и владений в рамках оценки уровня сформированности компетенции. При оценивании используется количественная шкала. Критерии оценивания приведены в таблице ниже.

Для оценивания результатов обучения на экзамене используются следующие содержательные показатели (формулируется с учетом конкретных требований дисциплины):

1. знание теоретических основ учебного материала, основных определений, понятий и используемой терминологии;
2. умение проводить обоснование и представление основных теоретических и практических результатов;
3. умение связывать теорию с практикой, иллюстрировать ответ примерами, в том числе, собственными, умение выявлять и анализировать основные закономерности, полученные, в том числе, в ходе выполнения лабораторных заданий;
4. умение обосновывать свои суждения и профессиональную позицию по излагаемому вопросу;
5. владение навыками программирования в рамках выполняемых лабораторных заданий.

Различные комбинации перечисленных показателей определяют критерии оценивания результатов обучения (сформированности компетенций) на государственном экзамене:

- высокий (углубленный) уровень сформированности компетенций;
- повышенный (продвинутый) уровень сформированности компетенций;
- пороговый (базовый) уровень сформированности компетенций.

Для оценивания результатов обучения на государственном экзамене используется 4-балльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Для оценивания результатов обучения на зачете используется – зачтено, не зачтено по результатам тестирования.

Соотношение показателей, критериев и шкалы оценивания результатов обучения на государственном экзамене представлено в следующей таблице.

Критерии оценивания компетенций и шкала оценок на экзамене

| Критерии оценивания компетенций | Уровень сформированности компетенций | Шкала оценок |
|--|--------------------------------------|-------------------|
| Обучающийся демонстрирует полное соответствие знаний, умений, навыков по приведенным критериям свободно оперирует понятийным аппаратом и приобретенными знаниями, умениями, применяет их при решении практических задач. Успешно выполнены лабораторные работы в соответствии с установленным перечнем. | Повышенный уровень | Отлично |
| Ответ на контрольно-измерительный материал не полностью соответствует одному из перечисленных выше показателей, но обучающийся дает правильные ответы на дополнительные вопросы. При этом обучающийся демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателям, но допускает незначительные ошибки, неточности, испытывает затруднения при решении практических задач. Успешно выполнены лабораторные работы в соответствии с установленным перечнем. | Базовый уровень | Хорошо |
| Обучающийся демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателям, допускает значительные ошибки при решении практических задач. При этом ответ на контрольно-измерительный материал не соответствует любым двум из перечисленных показателей, обучающийся дает неполные ответы на дополнительные вопросы. Успешно выполнены лабораторные работы в соответствии с установленным перечнем. | Пороговый уровень | Удовлетворительно |

| | | |
|--|---|---------------------|
| <p>Ответ на контрольно-измерительный материал не соответствует любым трем из перечисленных показателей. Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки. Не выполнены лабораторные работы в соответствии с установленным перечнем.</p> | - | Неудовлетворительно |
|--|---|---------------------|

Примерный перечень вопросов к экзамену

| № | Содержание |
|----|---|
| 1 | Технология разработки программ на языке C++ (создание исходного файла, компиляция, компоновка, отладка, тестирование, сопровождение) |
| 2 | Конструкции языка C++ |
| 3 | Объекты языка C++ |
| 4 | Динамическая память в C++, указатели, ссылки, динамические массивы |
| 5 | <i>Строки в языке C++</i> |
| 6 | <i>Структуры и объединения языка C++</i> |
| 7 | <i>Директивы препроцессора</i> |
| 8 | <i>Структурное программирование</i> |
| 9 | <i>Модульное программирование</i> |
| 10 | <i>Явное и неявное преобразование типов данных C++</i> |
| 11 | <i>Использование STL: контейнеры (vector, list), алгоритмы, итераторы</i> |
| 12 | <i>Базовые принципы объектно-ориентированного программирования</i> |
| 13 | <i>Инкапсуляция</i> |
| 14 | <i>Наследование</i> |
| 15 | <i>Полиморфизм</i> |
| 16 | <i>Конструкторы и деструкторы. Конструктор по умолчанию, копирующий конструктор, конструкторы в качестве операторов приведения типа</i> |
| 17 | <i>Разработка интерфейсов классов в C++ . Виртуальные функции, виртуальные деструкторы</i> |
| 18 | <i>Наследование. Виртуальные функции. Абстрактные классы</i> |
| 19 | <i>Перегрузка операторов и функций, дружественные функции и классы</i> |
| 20 | <i>Простое и множественное наследование, создание объектов производных классов при множественном наследовании</i> |
| 21 | <i>Спецификаторы наследования. Порядок вызова конструкторов и деструкторов. Передача параметров базовому классу при конструировании</i> |
| 22 | <i>Шаблоны классов</i> |

Пример контрольно-измерительного материала

УТВЕРЖДАЮ

Заведующий кафедрой технологий обработки и защиты информации

_____ А.А. Сирота

_____.2024

Направление подготовки / специальность 10.05.01 Компьютерная безопасность

Дисциплина Б1.В.ДВ.05.01 Разработка приложений на C++

Форма обучения Очное

Вид контроля Зачет с оценкой

Вид аттестации Промежуточная

Контрольно-измерительный материал № 1

1. Конструкторы и деструкторы. Конструктор по умолчанию, копирующий конструктор, конструкторы в качестве операторов приведения типа.
2. Базовые принципы объектно-ориентированного программирования.

Преподаватель _____ М.А. Дрюченко