

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Воронежский государственный технический университет»

УТВЕРЖДАЮ

Декан факультета информационных
технологий и компьютерной безопасности



/ П.Ю. Гусев /
И.О. Фамилия

«31» августа 2021 г.

РАБОЧАЯ ПРОГРАММА

дисциплины (модуля)

«Объектно-ориентированное программирование»

наименование дисциплины (модуля) в соответствии с учебным планом

Направление подготовки 09.03.01 Информатика и вычислительная техника

код и наименование направления подготовки/специальности

Профиль Системы автоматизированного проектирования

название профиля/программы

Квалификация выпускника бакалавр

Нормативный период обучения 4 г. / 4 г. 11 мес.

Очная/очно-заочная/заочная (при наличии)

Форма обучения Очная/Заочная

Год начала подготовки 2021 г.

Автор программы старший преподаватель _____

должность и подпись

А.А. Акинин

**Заведующий кафедрой
автоматизированных и
вычислительных систем** _____

подпись

В.Ф. Барабанов

Руководитель ОПОП _____

подпись

А.В. Бредихин

ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

формирование у обучающихся объектно-ориентированного мышления, изучение объектно-ориентированной методологии программирования, изучение ключевых понятий объектно-ориентированного программирования.

1.2. Задачи освоения дисциплины

– к теоретическим задачам относятся: ознакомление с тенденцией развития программного обеспечения; обучение разработке алгоритмов на основе объектно-ориентированного подхода; изучение основных технологий программирования C++, а также стандартов кодирования, спецификаций и последующих решений на их основе;

прикладные задачи состоят в ознакомлении с современными программными средами, приобретении навыков проектирования, разработки и сопровождения программ в рамках объектно-ориентированного стиля программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина (модуль) «Объектно-ориентированное программирование» относится к дисциплинам обязательной части блока Б.1 учебного плана.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Объектно-ориентированное программирование» направлен на формирование следующих компетенций:

ОПК-8 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

ОПК-9 - Способен осваивать методики использования программных средств для решения практических задач.

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-8	знать – язык программирования C++, операционную систему WINDOWS и оболочки, среду разработки программного обеспечения MS Visual Studio;
	уметь – составлять блок-схемы алгоритмов, разрабатывать алгоритмы с применением объектно-ориентированного подхода, писать и отлаживать коды на языке программирования C++, тестировать работоспособность программ в среде MS Visual Studio,

	использовать стандартные библиотеки языка C++;
	<p>владеть</p> <ul style="list-style-type: none"> – языком программирования C++; навыками отладки и тестирования программ в среде MS Visual Studio;
ОПК-9	<p>знать</p> <ul style="list-style-type: none"> – классификацию программных средств, порядок настройки и работы в интегрированных средах разработки программного обеспечения (на примере MS Visual Studio);
	<p>уметь</p> <ul style="list-style-type: none"> – находить в технической документации по использованию программного средства функции, необходимые для решения конкретной задачи;
	<p>владеть</p> <ul style="list-style-type: none"> – навыками разработки технической документации по использованию ПО для решения конкретной задачи; – навыками создания презентаций и видеороликов о методике использования программного средства для решения конкретной задачи.

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины «Объектно-ориентированное программирование» составляет 4 зачетные единицы.

Распределение трудоемкости дисциплины по видам занятий

Очная форма обучения

Вид учебной работы	Всего часов	Семестры
		2
Аудиторные занятия (всего)	90	90
В том числе:		
Лекции	36	36
Практические занятия (ПЗ)		
Лабораторные работы (ЛР)	54	54
Самостоятельная работа	54	54
Курсовой проект(работа) (есть, нет)	есть	есть
Контрольная работа(есть, нет)	нет	нет
Часы на контроль	-	-
Вид промежуточной аттестации (зачет, зачет с оценкой, экзамен)	зачет с оценкой	зачет с оценкой
Общая трудоемкость	144	144
	4 зач. ед.	4

Заочная форма обучения

Вид учебной работы	Всего часов	Семестры
		2
Аудиторные занятия (всего)	14	14
В том числе:		
Лекции	2	2
Практические занятия (ПЗ)		
Лабораторные работы (ЛР)	12	12
Самостоятельная работа	126	126
Курсовой проект(работа) (есть, нет)	есть	есть
Контрольная работа(есть, нет)	нет	нет
Вид промежуточной аттестации (зачет, зачет с оценкой, экзамен)	зачет с оценкой	зачет с оценкой
Часы на контроль	4	4
Общая трудоемкость	144	144
	4 зач. ед.	4

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

п/п	Наименование темы	Содержание раздела	Лекц.	Прак зан.	Лаб. зан.	СРС	Всего, час
1	Интегрированные среды и средства разработки на языке программирования высокого уровня	Введение. Анализ языков программирования и обзор сред разработки программного обеспечения. Возможности и производительность современных компиляторов, сборщиков проектов. Среды разработки. Интерфейс и конфигурация интегрированной среды разработки (IDE) под решение вычислительных задач. Разработка консольных проектов и приложений с графическим интерфейсом	8		12	10	30
2	Базовые конструкции и ключевые слова языка программирования высокого уровня	Синтаксис и операции. Типы данных языка и расширения типов, предоставляемые средой. Арифметические операции. Правила именования методов, переменных и констант. Основные алгоритмические конструкции. Использование структур, условных конструкций, циклов, функций, а также директив препроцессора.	8		12	10	30
3	Объектно-ориентированный подход к разработке программных средств	Введение в классы и объекты. Описание и свойства объектов. Конструкторы, деструкторы классов, объявления статических объектов, переменных и методов в классе. Наследование в ООП. Наследование, виды наследования. Абстрактные классы. Множественное наследование. Управление данными, директивы препроцессора. Структуры, исключения, набор директив для сборки проектов.	14		20	22	56
4	Создание подсистем на основе подключаемых библиотек	Динамически подключаемые источники данных. Работа с системными библиотеками операционных систем.	6		10	12	28
Итого			36		54	54	144

заочная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц.	Прак зан.	Лаб. зан.	СРС	Всего, час
1	Интегрированные среды и средства разработки на языке программирования высокого уровня	Введение. Анализ языков программирования и обзор сред разработки программного обеспечения. Возможности и производительность современных компиляторов, сборщиков проектов. Среды разработки. Интерфейс и конфигурация интегрированной среды разработки (IDE) под решение вычислительных задач. Разработка консольных проектов и приложений с графическим интерфейсом	0,5		2	25	27,5
2	Базовые конструкции и ключевые слова языка программирования высокого уровня	Синтаксис и операции. Типы данных языка и расширения типов, предоставляемые средой. Арифметические операции. Правила именования методов, переменных и констант. Основные алгоритмические конструкции. Использование структур, условных конструкций, циклов, функций, а также директив препроцессора.	0,5		3	25	28,5

3	Объектно-ориентированный подход к разработке программных средств	Введение в классы и объекты. Описание и свойства объектов. Конструкторы, деструкторы классов, объявления статических объектов, переменных и методов в классе. Наследование в ООП. Наследование, виды наследования. Абстрактные классы. Множественное наследование. Управление данными, директивы препроцессора. Структуры, исключения, набор директив для сборки проектов.	0,5		5	50	55,5
4	Создание подсистем на основе подключаемых библиотек	Динамически подключаемые источники данных. Работа с системными библиотеками операционных систем.	0,5		2	26	28,5
Итого			2		12	126	140

5.2 Перечень лабораторных работ

Лабораторная работа № 1. Знакомство со средой разработки Microsoft Visual Studio 2010. Создание консольных и графических приложений.

Лабораторная работа № 2. Изучение базовых инструкций языка C++. Особенности работы с массивами в языке C++.

Лабораторная работа № 3. Использование встроенных типов данных языка C++. Манипулирование основными системами счисления.

Лабораторная работа № 4. Символьные и строковые типы данных в языке C++. Основные функции работы со строками.

Лабораторная работа № 5. Работа с файлами в языке C++. Запись и чтение информации из файла.

Лабораторная работа № 6. Введение в понятие класса в языке C++.

Лабораторная работа № 7. Динамические структуры данных. Организация списка с помощью ООП.

Лабораторная работа № 8. Углубленное изучение концепций инкапсуляции, наследования и полиморфизма.

Лабораторная работа № 9. Изучение основных контейнеров и алгоритмов стандартной библиотеки шаблонов (STL).

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта во 2 семестре для очной формы обучения, во 2 семестре для заочной формы обучения.

Примерная тематика курсового проекта: «Применение объектно-ориентированной методологии при разработке программных модулей» (задания по вариантам).

Задачи, решаемые при выполнении курсового проекта:

- получение навыков применения основных технологий программирования C++;
- приобретении навыков проектирования, разработки и сопровождения программ в рамках объектно-ориентированного стиля программирования.
- практическое освоение современных программных сред.

Курсовой проект включают в себя программный продукт и расчетно-пояснительную записку.

Учебным планом по дисциплине «Объектно-ориентированное программирование» не предусмотрено выполнение контрольной работы.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ОПК-8	знать – язык программирования C++, операционную систему WINDOWS и оболочки, среду разработки программного обеспечения MS Visual Studio;	Тестирование Проверка этапов выполнения курсового проекта Ответы на теоретические вопросы при защите лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь – составлять блок-схемы алгоритмов, разрабатывать алгоритмы с применением объектно-ориентированного подхода, писать и отлаживать коды на языке программирования C++, тестировать работоспособность программ в среде MS Visual Studio, использовать стандартные библиотеки языка C++;	Эффективность использования изученного теоретического материала при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

	владеть – языком программирования C++; навыками отладки и тестирования программ в среде MS Visual Studio;	Владение языком программирования C++ при выполнении лабораторных работ Владение навыками отладки и тестирования работоспособности программы, реализованной на языке программирования C++, при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-9	знать – классификацию программных средств, порядок настройки и работы в интегрированных средах разработки программного обеспечения (на примере MS Visual Studio);	Тестирование Проверка этапов выполнения курсового проекта Ответы на теоретические вопросы при защите лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь – находить в технической документации по использованию программного средства функции, необходимые для решения конкретной задачи;	Эффективность использования изученного теоретического материала при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть – навыками разработки технической документации по использованию ПО для решения конкретной задачи; – навыками создания презентаций и видеороликов о методике использования программного средства для решения конкретной задачи.	Владение способами описания методики использования программного средства при выполнении лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются во 2 семестре для очной формы обучения, 2 семестре для заочной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-8	знать – язык программирования C++, операционную систему WINDOWS и оболочки.	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов

	<p>среду разработки программного обеспечения MS Visual Studio;</p> <p>уметь – составлять блок-схемы алгоритмов, разрабатывать алгоритмы с применением объектно-ориентированного подхода, писать и отлаживать коды на языке программирования C++, тестировать работоспособность программ в среде MS Visual Studio, использовать стандартные библиотеки языка C++;</p> <p>владеть – языком программирования C++; навыками отладки и тестирования программ в среде MS Visual Studio;</p>	<p>Решение стандартных практических задач</p> <p>Решение прикладных задач в конкретной предметной области</p>	<p>Задачи решены в полном объеме и получены верные ответы</p> <p>Задачи решены в полном объеме и получены верные ответы</p>	<p>Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах</p> <p>Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах</p>	<p>Продемонстрирован верный ход решения в большинстве задач</p> <p>Продемонстрирован верный ход решения в большинстве задач</p>	<p>Задачи не решены</p> <p>Задачи не решены</p>
ОПК-9	<p>знать – классификацию программных средств, порядок настройки и работы в интегрированных средах разработки программного обеспечения (на примере MS Visual Studio);</p> <p>уметь – находить в технической документации по использованию программного средства функции, необходимые для решения конкретной задачи;</p> <p>владеть – навыками разработки технической документации по использованию ПО для решения конкретной задачи; – навыками создания презентаций и видеороликов о методике использования программного средства для решения конкретной задачи.</p>	<p>Тест</p> <p>Решение стандартных практических задач</p> <p>Решение прикладных задач в конкретной предметной области</p>	<p>Выполнение теста на 90-100%</p> <p>Задачи решены в полном объеме и получены верные ответы</p> <p>Задачи решены в полном объеме и получены верные ответы</p>	<p>Выполнение теста на 80-90%</p> <p>Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах</p> <p>Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах</p>	<p>Выполнение теста на 70-80%</p> <p>Продемонстрирован верный ход решения в большинстве задач</p> <p>Продемонстрирован верный ход решения в большинстве задач</p>	<p>В тесте менее 70% правильных ответов</p> <p>Задачи не решены</p> <p>Задачи не решены</p>

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Выберите неверное высказывание?

- A. К элементам массива можно получить доступ с помощью арифметики указателей.
- B. Имя массива дает адрес первого элемента массива.
- C. Массив и указатель ничем не различаются. +
- D. Если массив передается в параметре функции, то он всегда передается как указатель

2. Правильно ли, что «Структура является собранием одной или нескольких переменных».

- A. Да +
- B. Нет
- C. Только нескольких

3. Верно ли, что «Для определения структуры нужно указать только данные для хранения переменных».

- A. Да+
- B. Нет
- C. Данные и их адрес

4. Что, помимо полей, могут включать в себя структуры?

- A. Указатель на ту же структуру +
- B. Заголовки функций +
- C. Еще одну структуру+
- D. Ничего

5. Чтобы обратиться к искомому элементу входящего в структуру массива нужно:

- A. Указать адрес массива
- B. Указать индекс
- C. Сделать все выше перечисленное +

6. Чем различаются операторы выбора «. » и «→» ?

- A. Оператор «→» применяется к объекту структурного типа, а оператор «. » к указателю на объект структурного типа.
- B. Операторы «. » и «→» применяются как к объекту структурного типа, так и к указателю на объект структурного типа.
- C. Оператор «. » применяется к объекту структурного типа, а оператор «→», к указателю на объект структурного типа.+

7. В программе описано класс и объект `class A {public: int a, b, c; }; A * obj;` Как обратиться к атрибуту `c`?
- A. `obj.c`
 - B. `obj→ c +`
 - C. `obj A → → c`
 - D. `obj→ A.c`
8. Свойство формы `name` – это...
- A. Имя формы, используется для управления формой и доступа к компонентам формы +
 - B. Текст заголовка (надпись на форме)
 - C. Значок в заголовке диалогового окна, обозначающий кнопку вывода системного меню
9. Какой оператор используется для динамического выделения массива объектов типа `T`?
- A. `allocate T`
 - B. `new T`
 - C. `new T[x]+`
10. Какое значение будет иметь `p` после выполнения следующего кода
- ```
int *p = 1000;
p++;
```
- A. 0
  - B. 1004+
  - D. 1001

### 7.2.2 Примерный перечень заданий для решения стандартных задач

1. Какова причина появления свободных участков памяти между соседними полями в структуре.
  - A. Из-за наличия требований выравнивания определенных типов; +
  - B. Если поле структуры является массивом;
  - C. Из-за не соблюдения правил описания полей структуры.
2. Для хранения значения переменной объединения выделяется памяти...
  - A. Столько, чтобы вместить самый "широкий" элемент объединения; +
  - B. Сумма значений занимаемых каждым элементом в отдельности;
  - C. Столько, чтобы вместить первый элемент объединения.
3. С помощью чего реализуется принцип полиморфизма в `C ++`?
  - A. Наличие множественного наследования.
  - B. Наличие виртуальных методов. +

- C. Использование виртуального наследования.
- D. Наличие абстрактных классов.

4. Компонент, предоставляющий возможность ввода данных в поле редактирования путем набора на клавиатуре или выбором из списка

- A. Button
- B. Memo
- C. CheckBox
- B. ComboBox +

5. Свойство, определяющее высоту объекта в пикселах.

- A. Height +
- B. Width
- C. Color

6. Элемент управления: `Image`

- A. Отображает графическое изображение на форме +
- B. Служит для отображения простейших графических объектов на форме: окружность, квадрат и т.п.
- C. Позволяет создать на форме прокручиваемую область с размерами большими, чем экран, на которой можно разместить объекты

7. Свойство компонента `TEdit`, указывающее цвет фона - ...

- A. Color+
- B. Text
- E. Height

8. Компоненты, которые видны во время работы приложения, с ними напрямую может взаимодействовать пользователь, называются...

- A. Визуальными +
- B. Невизуальными
- C. Активными

9. Выберите верное утверждение о деструкторе класса в C++.

- A. Деструктор принимает в качестве параметра адрес того объекта, который нужно уничтожить.
- B. Деструктор принимает в качестве параметра указатель `this`.
- C. Деструктор не содержит параметров. +
- D. Деструктор принимает в качестве параметра тот объект, который нужно уничтожить.

10. Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?

- A. Шаблонная.
- B. Полиморфная.

- C. Дружественная. +
- D. Статическая.

### 7.2.3 Примерный перечень заданий для решения прикладных задач

1. Какая из перечисленных функций не может быть конструктором?

- A. `void String () +`
- B. `String ();`
- C. `String (String & s)`
- D. `String (const int a)`

2. Если в программе на языке C ++ в производном классе переопределена операция `new` то ...

- A. Все объекты этого класса и все объекты классов, выведенных из него, будут использовать эту операцию независимо от зоне видимости, в которой она переопределена.
- B. Производные от этого класса могут использовать глобальную операцию применив операцию базовый\_класс :: `new`. +
- C. Операцию `new` нельзя переопределить.
- D. В любом случае эта операция будет доступна только в пределах класса-потомка.

3. Какой из перечисленных методов может быть конструктором для класса `String` в языке C ++?

- A. `String * String ();`
- B. `void String ();`
- C. `String (String & s); +`
- D. `const String (int a);`

4. Вызовет данный код ошибку компиляции? `class Rectangle { public: int a, b; int sum (); int square (); ~ Rect (); };`

- A. Ошибки нет, все записано верно.
- B. Ошибка являются: имя деструктора должно совпадать с именем класса. +
- C. Ошибка являются: имя деструктора не может начинаться с маленькой буквы.
- D. Ошибка являются: никакой идентификатор в C ++ не может начинаться со знака «~».

5. Укажите правильное объявление виртуального метода, который принимает одно целочисленное значение и возвращает `void`.

- A. `virtual void SomeFunction (int x); +`
- B. `void SomeFunction (int x) virtual;`
- C. `virtual SomeFunction (int x);`
- D. `virtual void SomeFunction (int * x);`

6. Укажите правильное использование оператора `friend`.

- A. class A {int\_friend CountPass (); private: short i;};
- B. class A {public: friend int H :: CountPass (); private: short i;};
- C. class A {public: int A1 :: CountPass (); friend: short i;};
- D. class A {public: friend int H :: q; short i;};

7. Принцип объектно-ориентированного программирования, заключающийся в объединении атрибутов и методов объекта с целью обеспечения сохранности данных, называется:

- A. Наследование.
- B. Сочетание.
- C. Инициализация.
- D. Инкапсуляция. +

8. В программе описаны абстрактный класс A и производный от этого класса класс A1. Какая из записей заведомо неверная?

- A. A \* a = new A; +
- B. A1 a1;
- C. A1 a1; A & a = a1;
- D. A1 a1; A1 a2;

9. Укажите правильный вариант доступа к членам объектов (язык C ++), описанных следующим образом: class my {char s; public: double Z; int f (int c, int d) {return c + d;}; } T1, T2;

- A. T1.Z = 23.1; +
- B. T2-> f (2,1);
- C. T1.s = '#';
- D. my.T2-> s = 'L';

10. Какой из вариантов записи абстрактного класса в C ++ является правильным?

- A. abstract class A {virtual int f () = 0;};
- B. class A {virtual int f () = 0;};
- C. class A {virtual int f () = 0;} abstract;
- D. class A {virtual int f ();};

#### 7.2.4 Примерный перечень вопросов для подготовки к зачету

1. Понятие класса в C++.
2. Перегруженные и встраиваемые (inline) функции в C++.
3. Области видимости объектов.
4. Пространства имен в языке C++.
5. Работа с шаблонными классами в C++.
6. Понятие пользовательского интерфейса, критерии и свойства при разработке приложений.
7. Понятие указателя и ссылки в языке C++.
8. Области видимости переменных.

9. Среда разработки (IDE) (работа с листингом, отладчиком, настройка и сборка проекта).
10. Динамически распределяемая память. Операторы new и delete.
11. Механизм обработки исключений.
12. Понятие конструктора и деструктора класса. Особенности их применения
13. Понятие виртуальных функций.
14. Основы стандартной библиотеки шаблонов. Список.
15. Введение в шаблонные функции.
16. Основные понятия ООП. Наследование.
17. Основные понятия ООП. Инкапсуляция.
18. Константные и статические переменные. Особенности модификаторов const и static.
19. Основы стандартной библиотеки шаблонов. Вектор.
20. Основные понятия ООП. Полиморфизм.
21. Основы стандартной библиотеки шаблонов. Очередь и двусторонняя очередь.
22. Понятие контейнерного класса. Понятие итератора.
23. Основы перегрузки операторов в языке C++.
24. Константные свойства и методы классов. Нюансы их использования.
25. Статические свойства и методы классов. Нюансы их использования.

### **7.2.5 Примерный перечень вопросов для подготовки к экзамену**

Не предусмотрено учебным планом

### **7.2.6 Методика выставления оценки при проведении промежуточной аттестации**

Оценка при проведении промежуточной аттестации учитывает результаты тестирования. Зачет проводится по теоретическим вопросам и практическим заданиям (один из п. 7.2.4 теоретический вопрос и одно практическое задание), тестирование предполагает получение ответов на 10 вопросов.

1. Оценка «Неудовлетворительно» ставится в случае, если студент дал неправильный ответ на теоретический вопрос, не решил практическое задание и ответил менее чем на 60% тестовых вопросов.

2. Оценка «Удовлетворительно» ставится в случае, если студент ответил на теоретический вопрос с ошибками и не решил полностью практическое задание и ответил на 60-80 % тестовых вопросов.

3. Оценка «Хорошо» ставится в случае, если студент правильно ответил на теоретический вопрос, решил полностью практическое задание, продемонстрировал понимание материала, но допустил незначительные ошибки, а выполнил тест на 80-90%.

4. Оценка «Отлично» ставится, если студент дал обоснованные, глубокий и правильный ответ на теоретический вопрос, решил полностью практическое задание, продемонстрировал понимание материала, а также выполнил тест на 90-100%.

Компетенции считаются сформированными, если в ходе изучения дисциплины выполнены и защищены курсовой проект и лабораторные работы. Выполненные лабораторные работы являются допуском к сдаче экзамена.

### 7.2.7 Паспорт оценочных материалов

| № п/п | Контролируемые разделы (темы) дисциплины                                              | Код контролируемой компетенции | Наименование оценочного средства                                           |
|-------|---------------------------------------------------------------------------------------|--------------------------------|----------------------------------------------------------------------------|
| 1     | Интегрированные среды и средства разработки на языке программирования высокого уровня | ОПК-8, ОПК-9                   | Тест, зачет с оценкой, защита лабораторных работ, защита курсового проекта |
| 2     | Базовые конструкции и ключевые слова языка программирования высокого уровня           | ОПК-8, ОПК-9                   | Тест, зачет с оценкой, защита лабораторных работ, защита курсового проекта |
| 3     | Объектно-ориентированный подход к разработке программных средств                      | ОПК-8, ОПК-9                   | Тест, зачет с оценкой, защита лабораторных работ, защита курсового проекта |
| 4     | Создание подсистем на основе подключаемых библиотек                                   | ОПК-8, ОПК-9                   | Тест, зачет с оценкой, защита лабораторных работ, защита курсового проекта |

### 7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методике выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется с использованием выданных задач на бумажном носителе. Время решения задач 20 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется с использованием выданных задач на бумажном носителе. Время решения задач 25 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Защита курсового проекта осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.



## **8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **8.1 Перечень учебной литературы, необходимой для освоения дисциплины**

1. Холопкина Л.В. Программирование на C++. Учебное пособие / Л.В. Холопкина. – Воронеж, ВГТУ. – 2011. – 184 с.

2. Воробьев Э.И. Объектно-ориентированное программирование : учеб. пособие. Ч.1. / Э.И. Воробьев - Воронеж : ГОУВПО "Воронежский государственный технический университет", 2007. - 100 с.

3. Лисицин Д.В. Объектно-ориентированное программирование [Электронный ресурс]: конспект лекций/ Д.В. Лисицин— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2010.— 88 с.— Режим доступа: <http://www.iprbookshop.ru/44970.html>.— ЭБС «IPRbooks».

4. Ганцева Е.А. Программирование на C/C++ в среде VISUAL STUDIO 2008: лабораторный практикум: учеб. пособие / Е.А. Ганцева. - Воронеж : ГОУВПО "Воронежский государственный технический университет", 2011. - 193 с.

5. Ганцева Е.А. Разработка приложений в Visual Studio: лабораторный практикум: учеб. пособие. / Е.А. Ганцева .- Воронеж : ФГБОУ ВО "Воронежский государственный технический университет", 2015. - 130 с.

6. Холопкина Л.В. Сборник задач по дисциплине “Объектно-ориентированное программирование” / Л.В. Холопкина, Н.И. Гребенникова, М.П. Носачева. – Воронеж: ВГТУ, – 2012.

7. Новиков П.В. Объектно-ориентированное программирование [Электронный ресурс]: учебно-методическое пособие к лабораторным работам/ П.В. Новиков— Электрон. текстовые данные.— Саратов: Вузовское образование, 2017.— 124 с.— Режим доступа: <http://www.iprbookshop.ru/64650.html>.— ЭБС «IPRbooks».

8. Организация самостоятельной работы обучающихся : методические указания для студентов, осваивающих основные образовательные программы высшего образования – бакалавриата, специалитета, магистратуры: методические указания / сост. В.Н. Почехихина, И.Н. Крючкова, Е.И. Головина, В.Р. Демидов; ФГБОУ ВО «Воронежский государственный технический университет». – Воронеж, 2020. – 14 с.

9. Методические рекомендации по выполнению курсовых работ и проектов для студентов направлений 09.03.01, 09.04.01 Информатика и вычислительная техника очной и заочной форм обучения / ФГБОУ ВПО «Воронежский государственный технический университет»; сост. Н.И. Гребенникова, В.В. Сафронов, А.М. Нужный, А.В. Барабанов, Воронеж, 2020. - 20 с.

### **8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю),**

**включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем**

**Лицензионное ПО:**

- Windows Professional 7 Single Upgrade MVL A Each Academic
- Microsoft Office Word 2007
- Microsoft Office Power Point 2007

**Свободно распространяемое ПО:**

- Microsoft Visual Studio Community Edition

**Отечественное ПО:**

- Яндекс.Браузер
- Архиватор 7z
- Astra Linux

**Ресурс информационно-телекоммуникационной сети «Интернет»:**

- Образовательный портал ВГТУ
- <http://www.edu.ru/>
- <https://metanit.com/>

**Информационно-справочные системы:**

- <http://window.edu.ru>
- <https://wiki.cchgeu.ru/>

**Современные профессиональные базы данных:**

- <https://proglib.io>
- <https://msdn.microsoft.com/ru-ru/>
- <https://docs.microsoft.com/>

**9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА**

Для проведения лекционных занятий необходима аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой.

Для проведения лабораторных работ необходима лаборатория с ПК, оснащенными программами для проведения лабораторного практикума и обеспечивающими возможность доступа к локальной сети кафедры и Интернет, из следующего перечня:

- 307 (Лаборатория микропроцессорной техники)
- 309 (Лаборатория телекоммуникационных систем)
- 311 (Лаборатория разработки программных систем)
- 320 (Лаборатория общего назначения)
- 322 (Лаборатория распределённых вычислений)
- 324 (Специализированная лаборатория сетевых систем управления (научно-образовательный центр «АТОС»))
- 325 (Лаборатория автоматизации проектирования вычислительных комплексов и сетей)

Лаборатории расположены по адресу: 394066, г. Воронеж, Московский проспект, 179 (учебный корпус №3).

## 10 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Объектно-ориентированное программирование» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на компьютерах, в соответствии с методиками, приведенными в указаниях к выполнению работ.

Большое значение по закреплению и совершенствованию знаний имеет самостоятельная работа студентов. Информацию о всех видах самостоятельной работы студенты получают на занятиях.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта студенты должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта. Освоение дисциплины оценивается на зачете.

| Вид учебных занятий                   | Деятельность студента                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Лекция                                | Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.           |
| Лабораторная работа                   | Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.                                                                                                                                       |
| Самостоятельная работа                | Самостоятельная работа студентов способствует глубокому усвоения учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: <ul style="list-style-type: none"><li>- работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;</li><li>- выполнение домашних заданий и расчетов;</li><li>- работа над темами для самостоятельного изучения;</li><li>- участие в работе студенческих научных конференций, олимпиад;</li><li>- подготовка к промежуточной аттестации.</li></ul> |
| Подготовка к промежуточной аттестации | Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации.                                                                                                                                                                                                                                                                                                                                                                                               |

