

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
Высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ
декан факультета _____ С.А.Баркалов
«31» августа 2021 года



РАБОЧАЯ ПРОГРАММА
дисциплины

«Основы программирования и алгоритмизации»

Направление подготовки 38.03.05 БИЗНЕС-ИНФОРМАТИКА

Профиль Информационные системы в бизнесе

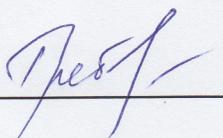
Квалификация выпускника бакалавр

Нормативный период обучения 4 года/4 года 11 м

Форма обучения очная/заочная

Год начала подготовки 2019

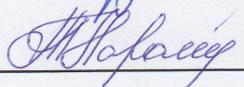
Автор программы

 /Гребенникова Н.И./

Заведующий кафедрой
автоматизированных и
вычислительных систем

 /Барабанов В.Ф./

Руководитель ОПОП

 /Наролина Т.С./

Воронеж 2021

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

состоит в обучении студентов практическим навыкам разработки алгоритмов решения инженерных и научных задач и их программной реализации с учетом современных тенденций развития вычислительной техники, информационных технологий в своей профессиональной деятельности, а также в изучении настройки системного, прикладного и инструментального программного обеспечения

1.2. Задачи освоения дисциплины

- изучение базовых алгоритмов решения вычислительных задач;
- освоение синтаксиса и базовых функций языка C++;
- изучения алгоритмов обработки сложных типов данных;
- освоение принципов объектно-ориентированного программирования;
- приобретение навыков программирования с использованием современных языков и программных средств.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Основы программирования и алгоритмизации» относится к дисциплинам базовой части блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Основы программирования и алгоритмизации» направлен на формирование следующих компетенций:

ОПК-3 - способностью работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-3	Знать - операционные системы и оболочки, современные среды разработки программного обеспечения; - основные методы разработки алгоритмов и программ на языке C++; - структуры данных, используемые для представления статических и динамических структур данных; - типовые алгоритмы обработки данных, в том числе методы сортировки
	Уметь

	- использовать стандартные библиотеки языка C++ для решения практических задач; - решать исследовательские и проектные задачи с использованием компьютеров
	Владеть методами построения современных проблемно-ориентированных прикладных программных средств на языке C++

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Основы программирования и алгоритмизации» составляет 5 з.е.

Распределение трудоемкости дисциплины по видам занятий
очная форма обучения

Виды учебной работы	Всего часов	Семестры
		1
Аудиторные занятия (всего)	72	72
В том числе:		
Лекции	36	36
Лабораторные работы (ЛР)	36	36
Самостоятельная работа	72	72
Курсовой проект	+	+
Часы на контроль	36	36
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость: академические часы	180	180
зач.ед.	5	5

заочная форма обучения

Виды учебной работы	Всего часов	Семестры
		1
Аудиторные занятия (всего)	22	22
В том числе:		
Лекции	8	8
Лабораторные работы (ЛР)	14	14
Самостоятельная работа	149	149
Курсовой проект	+	+
Часы на контроль	9	9
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость: академические часы	180	180
зач.ед.	5	5

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	Лаб. зан.	СРС	Всего, час
1	Основы С++	Базовые средства языка С++ Алфавит и лексемы языка. Типы данных С++. Структура программы. Переменные и выражения. Основные операторы языка С++ Базовые конструкции структурного программирования. Оператор «выражение». Операторы ветвления. Операторы цикла. Операторы передачи управления. Массивы. Одномерные массивы. Многомерные массивы. Строки и структуры. Строковые переменные. Структуры.	12	12	29	53
2	Динамическая память. Методы сортировки массивов	Указатели. Указатели. Операции с указателями. Динамические массивы. Генератор случайных чисел. Методы сортировки массивов. Сортировка методом пузырька. Сортировка методом поимка минимума (максимума). Сортировка методом вставки.	6	6	25	37
3	Функции	Функции. Объявление и определение функций. Возвращаемое значение и параметры функции. Передача массивов в качестве параметров. Рекурсивные функции. Передача имен функций в качестве параметров. Функция main(). Функции ввода/вывода Открытие потока. Ввод и вывод в поток. Закрытие потока. Обработка ошибок. Пример работы с файлами.	8	8	7	23
4	Динамические структуры данных	Динамические структуры данных Общие сведения. Линейные списки. Стеки. Очереди. Бинарные деревья. Пример реализации динамических структур данных Постановка задачи. Текст программы.	4	4	5	13
5	Основы объектно-ориентированного программирования	Объектно-ориентированное программирование. Классы ятия. Классы и их описания. Конструкторы и ические элементы класса. Наследование. кции и классы. Пример реализации класса Задание. Текст программы	6	6	6	18
Итого			36	36	72	144

заочная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Основы С++	<u>Самостоятельное изучение.</u> Базовые средства языка С++ Алфавит и лексемы языка. Типы данных С++. Структура программы. Переменные и выражения. Основные операторы языка С++ Базовые конструкции структурного программирования. Оператор «выражение». Операторы ветвления. Операторы цикла. Операторы передачи управления. <u>Самостоятельное изучение.</u> Массивы. Одномерные массивы. Многомерные массивы.	2	4	35	41

		<u>Самостоятельное изучение. Строки и структуры.</u> Строковые переменные. Структуры.				
2	Динамическая память. Методы сортировки массивов	Указатели. Указатели. Операции с указателями. Динамические массивы. Генератор случайных чисел. <u>Самостоятельное изучение. Методы сортировки массивов.</u> Сортировка методом пузырька. Сортировка методом поимка минимума (максимума). Сортировка методом вставки.	2	4	35	41
3	Функции	<u>Самостоятельное изучение.</u> Функции. Объявление и определение функций. Возвращаемое значение и параметры функции. Передача массивов в качестве параметров. Рекурсивные функции. Передача имен функций в качестве параметров. Функция main(). <u>Самостоятельное изучение.</u> Функции ввода/вывода Открытие потока. Ввод и вывод в поток. Закрытие потока. Обработка ошибок. Пример работы с файлами.		4	9	13
4	Динамические структуры данных	<u>Самостоятельное изучение.</u> Динамические структуры данных Общие сведения. Линейные списки. Стеки. Очереди. Бинарные деревья. Пример реализации динамических структур данных Постановка задачи. Текст программы.	2		35	37
5	Основы объектно-ориентированного программирования	<u>Самостоятельное изучение.</u> Объектно-ориентированное программирование. Классы Основные понятия. Классы и их описания. Конструкторы и деструкторы. Статические элементы класса. Наследование. Дружественные функции и классы. Пример реализации класса Задание. Текст программы	2	2	35	39
Итого			8	14	149	171

5.2 Перечень лабораторных работ

Неделя семестра	Наименование лабораторной работы	Объем часов	Виды контроля
1 семестр – очная форма обучения		36	
Основы C++		12	
1, 2	Лабораторная работа № 1 «Линейные и разветвляющиеся процессы»	4	Отчет, демонстрация работы на компьютере
3, 4	Лабораторная работа № 2 «Работа с одномерными и двумерными массивами»	4	Отчет, демонстрация работы на компьютере
5, 6	Лабораторная работа № 3 «Строки и структуры»	4	Отчет, демонстрация работы на компьютере
Динамическая память. Методы сортировки массивов		6	
7, 8, 9	Лабораторная работа № 4 «Указатели и динамические массивы»	6	Отчет, демонстрация работы на компьютере
Функции		8	
10, 11	Лабораторная работа № 5	4	Отчет, демонстрация

	«Функции»		работы на компьютере
12, 13	Лабораторная работа № 6 «Работа с потоками и файлами»	4	Отчет, демонстрация работы на компьютере
Динамические структуры данных		4	
14, 15	Лабораторная работа № 7 «Динамические структуры данных»	4	Отчет, демонстрация работы на компьютере
Основы объектно-ориентированного программирования		6	
16, 17, 18	Лабораторная работа № 8 «Объектно-ориентированное программирование. Создание и использование классов»	6	Отчет, демонстрация работы на компьютере
Итого часов на очной форме обучения		36	
1 семестр – заочная форма обучения		14	
Основы C++			
В сессию	Лабораторная работа № 1 «Линейные и разветвляющиеся процессы»	2	Отчет, демонстрация работы на компьютере
В сессию	Лабораторная работа № 2 «Работа с одномерными и двумерными массивами»	2	Отчет, демонстрация работы на компьютере
Динамическая память. Методы сортировки массивов		4	
В сессию	Лабораторная работа № 4 «Указатели и динамические массивы»	4	Отчет, демонстрация работы на компьютере
Функции		4	
В сессию	Лабораторная работа № 5 «Функции»	4	Отчет, демонстрация работы на компьютере
Основы объектно-ориентированного программирования			
В сессию	Лабораторная работа № 8 «Объектно-ориентированное программирование. Создание и использование классов»	2	Отчет, демонстрация работы на компьютере
Итого часов на заочной форме обучения		14	

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

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 1 семестре для очной формы обучения, в 1 семестре для заочной формы обучения.

Примерная тематика курсового проекта: «Разработка алгоритмического и программного обеспечения приложения, работающего с динамическими структурами данных»

Задачи, решаемые при выполнении курсового проекта:

– Создание алгоритмического и программного обеспечения приложения по обработке информации о наличии автобусов в автобусном парке

- Создание алгоритмического и программного обеспечения приложения по обработке информации о книгах в библиотеке
- Создание алгоритмического и программного обеспечения приложения по обработке информации о заявках на авиабилеты
- Создание алгоритмического и программного обеспечения приложения телефонного справочника
- Создание алгоритмического и программного обеспечения приложения по обработке информации о наличии бензина на автозаправочной станции
- Создание алгоритмического и программного обеспечения приложения по обработке информации о врачах, работающих в поликлинике
- Создание алгоритмического и программного обеспечения приложения по обработке информации о пациентах больницы
- Создание алгоритмического и программного обеспечения приложения по обработке информации для учета автомобильных грузоперевозок
- Создание алгоритмического и программного обеспечения приложения по обработке информации о наличии приборов на складе
- Создание алгоритмического и программного обеспечения приложения по обработке информации о заказах на ремонт вычислительной техники
- Создание алгоритмического и программного обеспечения приложения по обработке информации о лабораториях института
- Создание алгоритмического и программного обеспечения приложения по обработке информации о продаже мебельных гарнитур
- Создание алгоритмического и программного обеспечения приложения по обработке информации о продаже автомобилей в автосалоне
- Создание алгоритмического и программного обеспечения приложения по обработке информации о заказах на пошив одежды в ателье
- Создание алгоритмического и программного обеспечения приложения по обработке информации о продаже билетов на концерт
- Создание алгоритмического и программного обеспечения приложения по обработке информации о продаже путевок в туристическом агентстве
- Создание алгоритмического и программного обеспечения приложения по обработке информации о пользователях компьютерной сети
- Создание алгоритмического и программного обеспечения приложения по обработке информации о продажах компьютерных игр
- Создание алгоритмического и программного обеспечения приложения по обработке информации о гостиничных номерах
- Создание алгоритмического и программного обеспечения приложения по обработке информации о таксопарке

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

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-3	Знать - операционные системы и оболочки, современные среды разработки программного обеспечения; - основные методы разработки алгоритмов и программ на языке C++; - структуры данных, используемые для представления статических и динамических структур данных; - типовые алгоритмы обработки данных, в том числе методы сортировки	Тестирование Проверка этапов выполнения курсового проекта Результаты проведения коллоквиума 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 - использовать стандартные библиотеки языка C++ для решения практических задач; - решать исследовательские и проектные задачи с использованием компьютеров	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 методами построения современных проблемно-ориентированных прикладных программных средств на языке C++	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

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 1 семестре для очной формы обучения, 1 семестре для заочной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компе-	Результаты обучения, характеризующие	Критерии	Отлично	Хорошо	Удовл.	Неудовл.
--------	--------------------------------------	----------	---------	--------	--------	----------

тенция	сформированность компетенции	оценивания				
ОПК-3	Знать - операционные системы и оболочки, современные среды разработки программного обеспечения; - основные методы разработки алгоритмов и программ на языке С++; - структуры данных, используемые для представления статических и динамических структур данных; - типовые алгоритмы обработки данных, в том числе методы сортировки	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	Уметь - использовать стандартные библиотеки языка С++ для решения практических задач; - решать исследовательские и проектные задачи с использованием компьютеров	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	Владеть методами построения современных проблемно-ориентированных прикладных программных средств на языке С++	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

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: timage

- 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

. 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 является: имя деструктора должно совпадать с именем класса. +
- C. Ошибка является: имя деструктора не может начинаться с маленькой буквы.
- D. Ошибка является: 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

Не предусмотрено учебным планом

7.2.5 Примерный перечень вопросов для подготовки к экзамену

1. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти сумму положительных элементов в вещественных массивах A[8] и B[5].

2. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти сумму четных элементов в целочисленных массивах D[10] и T[8].

3. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти максимальные элементы в целочисленных массивах F[11] и S[7].

4. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество положительных элементов в вещественных массивах Q[10] и T[12].

5. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество элементов, равных значению, введенному пользователем, в целочисленных массивах S[7] и R[12].

6. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти произведение отрицательных элементов в целочисленных массивах D[11] и R[9].

7. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество нечетных элементов в целочисленных массивах P[10] и Q[6].

8. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество элементов, больших значения, введенного

пользователем, в вещественных массивах $A[10]$ и $D[8]$.

9. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество элементов кратных значению, введенному пользователем, в целочисленных массивах $G[9]$ и $K[10]$.

10. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество нулевых элементов в целочисленных массивах $V[7]$ и $Z[8]$.

11. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество элементов, входящих в интервал, заданный пользователем, в вещественных массивах $F[8]$ и $T[11]$.

12. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти сумму элементов на главных диагоналях целочисленных матриц $D[3,3]$ и $Y[4,4]$.

13. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество четных элементов на главных диагоналях целочисленных матриц $R[4,4]$ и $T[2,2]$.

14. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество отрицательных элементов выше главной диагонали вещественных матриц $A[3,3]$ и $Z[2,2]$.

15. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество положительных элементов ниже главной диагонали целочисленных матриц $W[4,4]$ и $Z[3,3]$.

16. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти максимальные элементы в вещественных матрицах $P[3,3]$ и $Q[2,2]$.

17. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти минимальные положительные элементы в целочисленных матрицах $R[4,4]$ и $K[2,2]$.

18. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество нулевых элементов на главных диагоналях целочисленных матриц $D[4,4]$ и $E[3,3]$.

19. Составить программу, реализующую решение определенной задачи с использованием функций:

– найти количество строк, не содержащих нулевых элементов, в целочисленных матрицах $D[4,4]$ и $T[3,3]$.

20. Составить программу, реализующую решение определенной задачи с

использованием функций:

– найти количество столбцов, не содержащих отрицательных элементов, в целочисленных матрицах $R[4,4]$ и $T[2,2]$.

21. Создать программу, которая формирует данные определенной структуры в виде двунаправленного линейного списка. Список должен являться объектом класса.

В классе должны присутствовать методы для реализации следующих операций: ввод и вывод списка, добавление и удаление элементов, поиск элементов по ключу.

Список состоит из следующих полей:

ФИО студента (ключ); Оценка по физике; Оценка по математике.

22. Создать программу, которая формирует данные определенной структуры в виде двунаправленного линейного списка. Список должен являться объектом класса.

В классе должны присутствовать методы для реализации следующих операций: ввод и вывод списка, добавление и удаление элементов, поиск элементов по ключу.

Список состоит из следующих полей:

Название прибора (ключ); Тип прибора; Количество на складе.

23. Создать программу, которая формирует данные определенной структуры в виде двунаправленного линейного списка. Список должен являться объектом класса.

В классе должны присутствовать методы для реализации следующих операций: ввод и вывод списка, добавление и удаление элементов, поиск элементов по ключу.

Список состоит из следующих полей:

ФИО работника (ключ); Должность; Заработная плата.

24. Создать программу, которая формирует данные определенной структуры в виде двунаправленного линейного списка. Список должен являться объектом класса.

В классе должны присутствовать методы для реализации следующих операций: ввод и вывод списка, добавление и удаление элементов, поиск элементов по ключу.

Список состоит из следующих полей:

Шифр группы (ключ); Факультет; Число студентов.

25. Создать программу, которая формирует данные определенной структуры в виде двунаправленного линейного списка. Список должен являться объектом класса.

В классе должны присутствовать методы для реализации следующих операций: ввод и вывод списка, добавление и удаление элементов, поиск элементов по ключу.

Список состоит из следующих полей:

Номер авиаполка (ключ); Количество самолетов; Количество боеготовых самолетов.

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

(Например: Экзамен проводится по тест-билетам, каждый из которых содержит 10 вопросов и задачу. Каждый правильный ответ на вопрос в тесте оценивается 1 баллом, задача оценивается в 10 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

7.2.6 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Основы C++	ОПК-3	Тест, экзамен, защита лабораторных работ, защита курсового проекта
2	Динамическая память. Методы сортировки массивов	ОПК-3	Тест, экзамен, защита лабораторных работ, защита курсового проекта
3	Функции	ОПК-3	Тест, экзамен, защита лабораторных работ, защита курсового проекта
4	Динамические структуры данных	ОПК-3	Тест, экзамен, защита лабораторных работ, защита курсового проекта
5	Основы объектно-ориентированного программирования	ОПК-3	Тест, экзамен, 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, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно

методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы, курсового проекта или отчета по всем видам практик осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Сергеев М.Ю. Программирование вычислительных задач на С++ Учебное пособие / М.Ю. Сергеев - Воронеж: ВГТУ. – 2015. – 90 с.

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

3. Ганцева Е.А. Разработка приложений в Visual Studio: лабораторный практикум: учеб. пособие / Е.А. Ганцева. – Воронеж: ВГТУ. – 130 с.

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

5. 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».

6. 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».

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Лицензионное ПО

LibreOffice

Microsoft Windows 7, Microsoft Visual Studio 2010, Microsoft Office 2013 (Word, Access, Excel, PowerPoint, Visio)

Ресурс информационно-телекоммуникационной сети «Интернет»

<http://www.edu.ru/>
Образовательный портал ВГТУ

Информационная справочная система

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

Современные профессиональные базы данных

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Лекционная аудитория и аудитории для практических занятий, оснащённые мультимедийным демонстрационным оборудованием (проектор, экран, звуковоспроизводящее оборудование), обеспечивающим демонстрацию мультимедиаматериалов.

Аудитории для лабораторных занятий, оснащенные компьютерами с лицензионным программным обеспечением с возможностью подключения к сети «Интернет» и доступом в электронную информационно образовательную среду университета.

Аудитории для самостоятельной работы, оборудованные техническими средствами обучения: персональными компьютерами с лицензионным программным обеспечением с возможностью подключения к сети «Интернет» и доступом в электронную информационно-образовательную среду университета

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

По дисциплине «Основы программирования и алгоритмизации» 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 лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

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

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить 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 усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
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. Данные перед экзаменом, экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.

6 Лист регистрации изменений

№ п/п	Перечень вносимых изменений	Дата внесения изменений	Подпись заведующего кафедрой, ответственной за реализацию ОПОП
1	Внесены изменения в рабочие программы дисциплин в части состава используемого лицензионного программного обеспечения, современных профессиональных баз данных и справочных информационных систем	31.08.2020	
2	Внесены изменения в рабочие программы дисциплин в части состава используемого лицензионного программного обеспечения, современных профессиональных баз данных и справочных информационных систем	31.08.2021	
3	Актуализирован перечень литературы, необходимой для освоения дисциплины	31.08.2021	