

Федеральное государственное бюджетное образовательное учреждение  
высшего образования «Саратовский государственный технический университет  
имени Гагарина Ю.А.»

Филиал федерального государственного бюджетного образовательного  
учреждения высшего образования  
«Саратовский государственный технический университет  
имени Гагарина Ю.А.» в г. Петровске



УТВЕРЖДАЮ  
Директор филиала СГТУ  
имени Гагарина Ю.А. в г.Петровске  
Е.А.Бесшапошникова  
«30» июня 2025 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ОЦЕНКЕ КАЧЕСТВА  
ОСВОЕНИЯ СТУДЕНТАМИ  
ПРОГРАММ ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА**

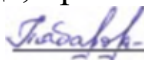
по дисциплине

ОП.04 «Основы алгоритмизации и программирования»

специальности

09.02.07 «Информационные системы и программирование»

Фонд оценочных средств рассмотрен  
на заседании предметной (цикловой) комиссии  
общепрофессиональных дисциплин и  
профессиональных модулей  
«16» июня 2025 года, протокол №13

Председатель ПЦК  /Ю.А.Табарова/

Петровск 2025

## **Пояснительная записка**

Фонд оценочных средств разработан в соответствии с рабочей программой ОП.04 «Основы алгоритмизации и программирования», требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 «Информационные системы и программирование», утверждённого приказом Министерства образования и науки РФ от 09.12.2016 г., № 1547 (ред. от 03.07.2024), ФГОС среднего общего образования и примерной основной образовательной программой.

### **1. Паспорт фонда оценочных средств для проведения текущего контроля успеваемости**

#### **1.1. Цели и задачи контроля**

Целью текущего контроля успеваемости обучающихся является обеспечение систематического контроля и оценки уровня освоения предметных результатов, уровня сформированности общих компетенций ОП.04 «Основы алгоритмизации и программирования».

Главной задачей текущего контроля успеваемости является повышение мотивации обучающихся к регулярной учебной и самостоятельной работе, закрепление, углубление знаний, закрепление и совершенствование умений, обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности посредством внедрения эффективной системы оценки в образовательный процесс.

#### **Предметные результаты:**

В рамках программы учебной дисциплины обучающимися осваиваются следующие **знания**:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- классификацию языков программирования; понятие системы программирования;
- основные элементы языка, структура программы;
- методы реализации типовых алгоритмов;
- операторы и операции, управляющие структуры, структуры данных, классы памяти;
- понятие подпрограммы, библиотеки подпрограмм;
- объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения.

В рамках программы учебной дисциплины обучающимися осваиваются следующие **умения**:

- разрабатывать и анализировать алгоритмы для решения поставленных задач;
- определять сложность алгоритмов;

- реализовывать типовые алгоритмы в виде программ на актуальных языках программирования;
- использовать средства проектирования для создания и графического отображения алгоритмов;
- оформлять код программ в соответствии со стандартом кодирования;
- выполнять проверку, отладку кода программы

**Общие компетенции, включающие в себя способность:**

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках

## **1.2. Структура фонда оценочных средств**

Фонд оценочных средств для проведения текущего контроля включает в себя комплекты контрольно-оценочных средств, предназначенные для проведения текущего контроля в виде:

- оперативного контроля;
- рубежного контроля.

Оперативный контроль проводится в форме:

- опрос (устный, письменный);
- тестирование;
- выполнение практической работы;

Рубежный контроль проводится в форме:

- опрос (устный, письменный);
- тестирование;
- выполнение практической работы;

Фонд оценочных средств также, включает в себя комплект контрольно-оценочных средств для проведения промежуточной аттестации. Межсессионная аттестация проводится в форме тестирования.

## **1.3. Система оценивания результатов выполнения заданий**

Оценивание результатов выполнения заданий промежуточной аттестации осуществляется на основе следующих принципов:

- достоверности оценки – оценивается уровень сформированности знаний, умений, практического опыта, общих и профессиональных компетенций, продемонстрированных обучающимися в ходе выполнения задания;
- адекватности оценки – оценка выполнения заданий должна проводиться в отношении тех компетенций, которые необходимы для эффективного

выполнения задания; надежности оценки – система оценивания выполнения заданий должна обладать высокой степенью устойчивости при неоднократных оценках уровня сформированности знаний, умений, практического опыта, общих и профессиональных компетенций обучающихся;

– комплексности оценки – система оценивания выполнения заданий должна позволять интегративно оценивать общие и профессиональные компетенции обучающихся;

– объективности оценки – оценка выполнения конкурсных заданий должна быть независимой от особенностей профессиональной ориентации или предпочтений преподавателей, осуществляющих контроль или аттестацию.

При выполнении процедур оценки заданий используются следующие основные методы:

– метод расчета первичных баллов;

– метод расчета сводных баллов.

Результаты выполнения заданий оцениваются в соответствии с разработанными критериями оценки. Используется пятибалльная шкала для оценивания результатов обучения.

Перевод пятибалльной шкалы учета результатов в пятибалльную оценочную шкалу:

Оценка	Количество баллов, набранных за выполнение теоретического и практического задания, средний балл по итогам аттестации
Оценка 5 «отлично»	4,6-5
Оценка 4 «хорошо»	3,6-4,5
Оценка 3 «удовлетворительно»	3-3,5
Оценка 2 «неудовлетворительно»	≤ 2,9

#### **1.4. Учебно-методическое и информационное обеспечение для проведения контроля**

##### **Основные учебные издания:**

1. Лебеденко, Л. Ф. Технологии программирования: учебно-методическое пособие для СПО / Л. Ф. Лебеденко, О. И. Моренкова. — 2-е изд. — Саратов: Профобразование, 2024. — 108 с. — ISBN 978-5-4488-1204-0. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/139115>

2. Медведев, М. А. Программирование на СИ#: учебное пособие для СПО / М. А. Медведев, А. Н. Медведев; под редакцией А. В. Присяжного. — 3-е изд. — Саратов, Екатеринбург: Профобразование, Уральский федеральный университет, 2024. — 62 с. — ISBN 978-5-4488-0471-7, 978-5-7996-2833-8. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/139593>

### **Дополнительные учебные издания:**

3. Биллиг, В. А. Основы программирования на C#: учебное пособие / В. А. Биллиг. — 4-е изд. — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 573 с. — ISBN 978-5-4497-0893-9. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/146368>
4. Зыков, С. В. Введение в теорию программирования. Объектно-ориентированный подход: учебное пособие для СПО / С. В. Зыков. — 2-е изд. — Саратов: Профобразование, 2024. — 187 с. — ISBN 978-5-4488-0995-8. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/139748>
5. Маляров, А. Н. Объектно-ориентированное программирование: учебник для СПО / А. Н. Маляров. — 2-е изд. — Саратов: Профобразование, 2022. — 334 с. — ISBN 978-5-4488-1561-4. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/132418>

### **Интернет-ресурсы:**

#### **Электронно-библиотечная система:**

6. ЭБС «Znanium»
7. ЭБС «PROФобразование»
8. ЭБС «Book.ru»

## **2. Контрольно-оценочные средства**

### Теоретическое занятие 1. Языки программирования

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

#### **Вопросы:**

1. Что такое язык программирования?
2. Что определяет язык программирования?
3. На какие группы можно разделить множество языков программирования?
4. Сколько языков программирования придумано в настоящее время?
5. Какие языки программирования являются языками высокого уровня?
6. Какие языки программирования являются языками низкого уровня?

### Теоретическое занятие 2. Языки программирования

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

#### **Вопросы:**

1. Что такое компилятор?
2. Что называется интерпретатором?
3. Чем отличается компилятор от интерпретатора?
4. Как работает компилятор?
5. Как работает интерпретатор?
6. Для чего нужны отладчики?

Теоретическое занятие 3. Жизненный цикл программы. Основные этапы решения задач на компьютере

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 20 мин;
- 3) максимальный балл за задание – 5 баллов

#### **Вопросы:**

1. Что такое жизненный цикл программы?
2. Какие этапы включает жизненный цикл программы?
3. Какие вопросы нужно решить на этапе анализа?
4. Что происходит на этапе определения спецификаций?
5. Что происходит на этапе проектирования?

6. Какие действия входят в этап постановки задачи?
7. Что происходит на этапе разработки алгоритма?
8. Что происходит на этапе программирования?
9. Что происходит на этапе тестирования и отладки?
10. Что происходит на этапе сопровождения программы?

#### Практическая работа 1: Знакомство со средой программирования

**Форма контроля:** Рубежный контроль

**Задание:**

1. Загрузить установщик программы
2. Установить программу
3. Настроить программу
4. Создать первый проект

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Самостоятельная работа 1. Классификация языков программирования

**Форма контроля:** Рубежный контроль

**Задание:**

Подготовить сравнительную таблицу языков программирования:

Название языка программирования	Год появления, разработчик	Описание языка	Достоинства	Недостатки

Порядок выполнения работы согласно методическим рекомендациям по самостоятельным работам по ОП.04 «Основы алгоритмизации и программирования».

#### Теоретическое занятие 4. Типы данных

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое тип данных? Для чего он применяется?
2. Какие типы данных вам известны?
3. Для чего предназначен символьный тип данных?
4. Для чего предназначен логический тип данных?
5. Чем символьный тип данных отличается от строкового?
6. Для чего предназначен файловый тип данных?

Теоретическое занятие 5. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое выражение в C#? Как в нём участвуют операнды, объединённые знаками операций?
2. Какие бывают операции в C# по количеству участвующих в одной операции операндов?
3. Что такое операнды простейшего выражения: константы, переменные и вызовы функций?
4. Какие есть операторы ввода и вывода в консольном приложении на языке C# и к какому классу они относятся?
5. Какие методы используют для ввода и вывода данных?
6. Что такое базовая операция присваивания в C# и как она работает?
7. Какие ещё есть операции присваивания в C#?
8. Как набор инструкций может объединяться в блок кода: в какие скобки заключается, между какими помещается?
9. Что такое операнды простейшего выражения: константы, переменные и вызовы функций?
10. Что такое составной оператор?

Практическая работа 2: Составление программ линейной структуры

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать программу, которая выводит на экран свою фамилию, имя, отчество и дату рождения
2. Используя стандартные математические операции, вычислить сумму, разность, произведение двух чисел

$$y = \sqrt{\frac{x+3}{x-3}}.$$

3. Написать программу для расчета функции

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 6. Условный оператор. Оператор выбора

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**



- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое условный оператор?
2. Какие операторы относятся к условным в C#?
3. Как работает условный оператор if?
4. Какие формы записи есть у условного оператора if?

Теоретическое занятие 7. Условный оператор. Оператор выбора

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое оператор выбора?
2. Как работает оператор switch?
3. Какие типы выражений допустимы в операторе switch?
4. Какие ограничения есть у оператора switch?

Практическая работа 3: Составление программ разветвляющей структуры

**Форма контроля:** Рубежный контроль

**Задание:**

1. Напишите консольную программу, в которую пользователь вводит с клавиатуры два числа. А программа сравнивает два введенных числа и выводит на консоль результат сравнения (два числа равны, первое число больше второго или первое число меньше второго).

2. В банке в зависимости от суммы вклада начисляемый процент по вкладу может отличаться. Напишите консольную программу, в которую пользователь вводит сумму вклада. Если сумма вклада меньше 100, то начисляется 5%. Если сумма вклада от 100 до 200, то начисляется 7%. Если сумма вклада больше 200, то начисляется 10%. В конце программа должна выводить сумму вклада с начисленными процентами

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 4: Составление программ разветвляющей структуры

**Форма контроля:** Рубежный контроль

**Задание:**

1. Выдать введенное число в словесной интерпретации
2. Напишите консольную программу, которая выводит пользователю сообщение "Введите номер операции: 1.Сложение 2.Вычитание 3.Умножение".

Рядом с названием каждой операции указан ее номер, например, операция вычитания имеет номер 2. Пусть пользователь вводит в программу номер операции, в зависимости от номера операции программа выводит ему название операции и с введенными числами выполняются определенные действия (например, при вводе числа 3 числа умножаются).

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 8. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое цикл?
2. Что называется телом цикла?
3. Назовите виды циклов?
4. Какой оператор отвечает за цикл с предусловием? Сколько раз такой цикл может выполняться?
5. Какой оператор отвечает за цикл с постусловием? Сколько раз такой цикл может выполняться?

Теоретическое занятие 9. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое цикл с параметром?
2. Чем отличаются for и while?
3. Что такое вложенный цикл?
4. В каком случае можно применять циклы в программе?

Практическая работа 5: Составление программ циклической структуры

**Форма контроля:** Рубежный контроль

**Задание:**

1. Найти сумму первых N членов арифметической прогрессии с использованием цикла for

2. Найти среднее арифметическое всех целых чисел от  $a$  до  $b$  (значения  $a$  и  $b$  вводятся с клавиатуры)

3. За каждый месяц банк начисляет к сумме вклада 10% от суммы. Напишите консольную программу, в которую пользователь вводит сумму вклада и количество месяцев. А банк вычисляет конечную сумму вклада с учетом начисления процентов за каждый месяц

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Практическая работа 6: Составление программ циклической структуры

**Форма контроля:** Рубежный контроль

**Задание:**

1. Получить таблицу температур по Цельсию  $t_c$  от -50 до +50 градусов а также их эквивалентов по шкале Фаренгейта  $t_f$ , используя соотношение

2. Задания самостоятельного решения

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Теоретическое занятие 10. Массивы. Одномерные и двумерные массивы

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое одномерный массив?
2. Как объявить массив?
3. Как инициализировать массив?
4. Как обратиться к элементу массива?

#### Теоретическое занятие 11. Массивы. Одномерные и двумерные массивы

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое двумерный массив?
2. Как объявить двумерный массив?
3. Как инициализировать двумерный массив?
4. Как передать двумерный массив в функцию?

### Практическая работа 7: Обработка одномерных массивов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Ввести размер массива и его элементы с клавиатуры. Найти сумму всех элементов
  2. Создать массив из 10 целых чисел. Заполнить его случайными числами от 1 до 100. Найти сумму всех элементов
  3. Ввести размер массива и его элементы. Найти максимальный элемент
- Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 8: Обработка одномерных массивов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Ввести размер массива и его элементы. Посчитать количество четных чисел.
  2. Дан массив, состоящий из 15 элементов целого типа. Получить новый массив, как разность между элементами исходного массива и его среднего арифметического
- Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 9: Обработка двумерных массивов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать программу, которая создает двумерный массив 3x3, заполняет его случайными числами от 1 до 100 и выводит на консоль
  2. Найти максимальный элемент в двумерном массиве и его позицию
- Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 10: Обработка двумерных массивов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать программу, которая транспонирует квадратную матрицу
  2. Найти сумму элементов главной и побочной диагоналей квадратной матрицы
- Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 12. Строки. Стандартные процедуры и функции для работы со строками

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Как в переменную типа string занести значение строки?
2. Как определить, равны ли две строки типа string между собой?
3. Как сравнить две строки типа string в лексикографическом порядке?
4. Как соединить две строки типа string?
5. Как скопировать одну строку типа string в другую?
6. Как вставить подстроку начиная из заданного индекса?

Теоретическое занятие 13. Строки. Стандартные процедуры и функции для работы со строками

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Как найти и получить индекс первого вхождения подстроки в данную строку?
2. Как найти и получить индекс последнего вхождения подстроки в данную строку?
3. Как определить длину строки типа string?
4. Как создать строку заданной ширины?
5. Как удалить заданное количество символов из строки?
6. Как заменить символы в строке?
7. Как в строке выделить подстроку?

Практическая работа 11: Работа со строками

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать программу, для определения – является ли введенное слово палиндромом
2. Написать программу, которая определяет совпадение подстроки в начале и в конце строки
3. Написать программу для вставки одной подстроки в другую строку. Строка и подстрока вводятся с клавиатуры

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 14. Структурированный тип данных – множество.

Операции над множествами

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое множество в С#?
2. Какие операции можно выполнять над множествами в С#?
3. Как происходит объединение множеств в С#?

Теоретическое занятие 15. Структурированный тип данных – множество.

Операции над множествами

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Как происходит вычисление разности множеств в С#?
2. Как происходит пересечение множеств в С#?
3. Как происходит проверка на подмножество в С#?

Практическая работа 12: Работа с данными типа множество

**Форма контроля:** Рубежный контроль

**Задание:**

1. Сформировать множество, содержащее 5 целых чисел из диапазона 1..10 и вывести это множество на экран
2. Дана строка. Содержит ли она и строчные символы?
3. Дана строка символов. Требуется построить и вывести множество, элементами которого являются гласные буквы, которые входят в текст

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 16. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа. Организация доступа к файлам

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое комбинированный тип данных запись?
2. Какие задачи связаны с этим понятием?
3. Какие классы определены для работы с текстовыми файлами?

Теоретическое занятие 17. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа. Организация доступа к файлам

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. В чём заключается достоинство последовательного метода доступа, как он может быть организован?
2. Что означает прямой способ доступа к записям, в чём его универсальность, какие требования к языку программирования для его реализации?
3. Какие файловые структуры используются для хранения данных во внешней памяти, в чём заключаются различия между файлами прямого и последовательного доступа?

Практическая работа 13: Составление программ с использованием текстовых файлов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать файл и записать в него текст
2. Чтение из файла
3. Добавить текст в конец файла
4. Записать массив строк в файл и прочитать его

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 14: Создание программ с использованием типизированных и нетипизированных файлов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Работа с типизированным файлом
2. Работа с нетипизированными файлами в C#

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Теоретическое занятие 18. Процедуры и функции

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

##### **Вопросы:**

1. Что такое подпрограмма?
2. Какие виды подпрограмм используются в программах?
3. Что такое процедура?
4. Что такое функция?

#### Теоретическое занятие 19. Процедуры и функции

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

##### **Вопросы:**

1. В чём основное отличие процедуры от функции в C#?
2. Когда целесообразно создавать функцию, а когда процедуру?
3. Почему функции не могут существовать в отрыве от классов в C#?
4. Что такое параметр и аргумент в контексте процедур и функций C#?

#### Практическая работа 15: Организация процедур

**Форма контроля:** Рубежный контроль

**Задание:**

1. В программе необходимо добавить вызовы процедуры таким образом, чтобы при вводе значения 0 выводилась на экран ошибка "Error: division by zero!", а при вводе любого другого числа выводилась ошибка "Error in input!". Ваша задача - оформить правильный вызов процедуры. Запрещенные операторы: return.

2. Напишите процедуру, которая выводит прямоугольный треугольник, на сторонах которого n символов 'm'

3. Напишите процедуру, которая выводит на экран все делители числа N в одну строку через пробел.



4. Напишите процедуру, которая принимает два параметра – натуральное число  $N$ , и строку  $S$  – и выводит на экран строку  $S$ , повторив ее  $N$  раз.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Практическая работа 16: Организация функций

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создайте метод `Sum ( )`, который принимает два целочисленных аргумента и суммирует их. Метод не возвращает никакого значения (именно поэтому вы должны использовать ключевое слово `void`).

2. Задания для самостоятельного решения

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 20. Рекурсия. Программирование рекурсивных алгоритмов

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое рекурсия в программировании?
2. Зачем нужна рекурсия?
3. Как работает рекурсивный алгоритм?
4. Какие условия должны быть в рекурсивном алгоритме?

#### Практическая работа 17: Применение рекурсивных функций

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать программу для вывода  $n$  первых членов арифметической прогрессии 1, 2, 3... с использованием рекурсивного метода

2. Напишите программу для поиска индекса максимального элемента массива с использованием рекурсии

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Теоретическое занятие 21. Структуризация в программировании

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое структурное программирование?
2. Что такое структура в С# и для чего её используют?
3. Какая общая форма объявления структуры (структурного типа)?
4. Какой тип доступа по умолчанию имеют поля структурной переменной?
5. Какие элементы языка программирования можно реализовывать в структуре?

Теоретическое занятие 22. Структуризация в программировании

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое структурное программирование?
2. Назовите принципы структурного программирования?
3. Назовите базовые конструкции структурного программирования?
4. Назовите методы структурного программирования.

Теоретическое занятие 23. Модульное программирование

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое модульное программирование?
2. Как большая задача разбивается на несколько меньших, логически завершённых и взаимосвязанных частей — модулей?
3. Назовите свойства модулей.

Теоретическое занятие 24. Модульное программирование

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое модуль?
2. Назовите структуру модуля.
3. Из каких файлов состоит модуль?
4. Как внутренние данные модуля могут быть скрыты от прямого доступа?
5. Назовите стандартные модули. Для чего они используются?

Практическая работа 18: Программирование модуля**Форма контроля:** Рубежный контроль**Задание:**

1. Написать программу для определения класса, описывающего некоторого студента вуза
2. Задания для самостоятельного решения.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 25. Указатели**Форма контроля:** оперативный контроль.**Задание:** ответить на вопросы устно (фронтальный опрос).**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое указатель?
2. Для чего применяются указатели?
3. Как объявить указатель?
4. Как инициализировать указатель?

Теоретическое занятие 26. Указатели**Форма контроля:** оперативный контроль.**Задание:** ответить на вопросы устно (фронтальный опрос).**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Какие операции можно производить с указателями?
2. Что такое разыменование указателя?
3. Как указатели позволяют напрямую обращаться к памяти?

Практическая работа 19: Использование указателей для организации связанных списков

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать список
2. Создать связный мультисписок на основе указателей.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 27. Основные принципы объектно-ориентированного программирования

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое объект?
2. Назовите его методы и свойства?
3. Что такое класс?
4. Что такое компоненты?

Теоретическое занятие 28. Основные принципы объектно-ориентированного программирования

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 15 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Назовите основные принципы ООП.
2. Что такое инкапсуляция в ООП и как она реализуется в C#?
3. Как инкапсуляция защищает данные объекта от некорректного доступа и изменения извне?
4. Что такое наследование в ООП и как оно позволяет создавать классы с общим функционалом, не копируя каждый раз одни и те же поля и методы?
5. Что такое полиморфизм в ООП и как он даёт возможность использовать одни и те же методы для объектов разных классов?
6. Что такое абстракция в ООП и как она достигается с помощью классов и интерфейсов, которые определяют, какие свойства и методы будут доступны для использования?
7. Как абстракция позволяет выделить общие характеристики объектов и скрыть их внутренние детали?

Теоретическое занятие 29. Интегрированная среда разработчика

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое интегрированная среда разработчика?
2. Приведите примеры интегрированных сред разработчика.
3. Что такое Visual Studio?
4. Какие инструменты для отладки и тестирования кода доступны в Visual Studio?
5. Какие типы проектов можно создать в Visual Studio?

#### Теоретическое занятие 30. Интегрированная среда разработчика

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Какие окна и инструменты входят в интерфейс Visual Studio?
2. Как настроить среду разработки для работы с проектом?
3. Какие ресурсы доступны для изучения интегрированной среды разработки Visual Studio?

#### Практическая работа 20: Изучение интегрированной среды разработчика

**Форма контроля:** Рубежный контроль

**Задание:**

1. Настроить форму
2. Размещение элементов управления.
3. Размещение строки ввода
4. Размещение надписей
5. Разместите на форме четыре кнопки (Button). Сделайте на кнопках следующие надписи: «красный», «зеленый», «синий», «желтый»

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

#### Самостоятельная работа 2. Виды сред разработки программ

**Форма контроля:** Рубежный контроль

**Задание:**

Подготовить презентацию, выбрав любую среду разработки программ

Порядок выполнения работы согласно методическим рекомендациям по самостоятельным работам по ОП.04 «Основы алгоритмизации и программирования».

Теоретическое занятие 31. Визуальное событийно-управляемое программирование

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое визуальное программирование и в чём его преимущество перед традиционным текстовым программированием?
2. Что такое событийно-управляемое программирование и на чём оно основано?
3. Какие действия пользователя приводят к возникновению соответствующих событий у программного объекта?
4. Какие действия может предусмотреть программист в обработке события?
5. Какие технологии программирования предоставляет среда Visual Studio при разработке Windows-приложений?

Теоретическое занятие 32. Визуальное событийно-управляемое программирование

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое элементы управления?
2. Назовите элементы управления, которые можно добавлять на формы.
3. Назовите основные свойства элементов управления.

Практическая работа 21: Создание проекта с использованием компонентов для работы с текстом

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написание программы обработки события нажатия кнопки
2. Написание программы обработки события загрузки формы.
3. Составить программу вычисления для заданных значений  $x$ ,  $y$ ,  $z$  арифметического выражения

4. Разработать программу с помощью, которой пользователь, введя свой рост и фактический вес, мог бы определить худой он или полный, и на сколько ему нужно поправиться или похудеть. Для разработки программы воспользуйтесь тем, что оптимальный вес человека определяется так: рост человека минус 100. Если фактический вес человека меньше оптимального, то человек худой, и наоборот, если больше, то нужно похудеть.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 22: Создание проекта с использованием компонентов для работы с текстом

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разместите на форме две кнопки (Button) и одну метку (Label). Сделайте на кнопках следующие надписи: «привет», «до свидания». Создайте обработчики события нажатия на данные кнопки, которые будут менять текст метки на слова, написанные на кнопках. Создайте обработчик события создания формы (Load), который будет устанавливать цвет формы и менять текст метки на строку «Начало работы».

2. Разместите на форме ряд кнопок (Button) напротив каждой поле ввода (TextBox) и одну метку (Label). Создайте обработчики события нажатия на данные кнопки, которые будут менять текст в метке. Текст в метке берется из поля ввода напротив нажимаемой кнопки.

3. Разработка проекта, содержащего форму, на которую помещены две кнопки и три элемента управления label. При нажатии на первую кнопку в элементе управления label1 выдается ваша фамилия, в label2 – имя, в label3 – отчество, при нажатии на вторую – надписи меняются местами.

Разработка проекта, содержащего форму, на которую помещена

4. кнопка управления, label и 2 элемента textBox. При нажатии на кнопку содержимое элементов управления textBox1 и textBox2 выдается в элементе управления label1.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 23: Создание проекта с использованием компонентов для работы с текстом

**Форма контроля:** Рубежный контроль

**Задание:**

1. Ввести три числа – x, y, z. Вычислить

2. Разработать программу, которая при нажатии на кнопку «Нажать» выводит сообщение «Я студент специальности Информационные системы и программирование», а затем при повторном нажатии на эту же кнопку

сообщение исчезает. При повторном выводе цвет надписи должен принять любой цвет, кроме черного.

3. Разработка проекта, содержащего форму, на которую помещена кнопки и элементы управления label. При нажатии на одну кнопку меняется цвет фона, а на другую – цвет текста в label

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 24: Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать календарь
2. Извлечение диапазона дат из элемента управления календарем месяца.
3. Форматирование даты в метке
4. Отобразить несколько месяцев в календаре

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 25: Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать программу Birthday
2. Реализовать программу, которая будет узнавать текущую дату и время.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 26: Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разработка проекта, содержащего форму, на которую помещена кнопка и элементы управления textBox и dateTimePicker. При нажатии на кнопку содержимое элемента управления dateTimePicker выдается в textBox.

2. Разработка проекта, содержащего форму, на которую помещена кнопка, элементы управления label и monthCalendar. При нажатии на кнопку в элементе управления label1 выдается выбранная дата, а в label2 день недели.

3. Разработка проекта, содержащего форму, на которую помещена кнопка и элементы управления label, в котором содержится слово «сегодня» и dateTimePicker. При нажатии на кнопку по выбранной дате в dateTimePicker в label1 добавляется день недели, а в label2 текущее время.



Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 27: Создание проекта с использованием полос прокрутки для ввода информации

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создание полос прокруток на форме
2. Создайте приложение, на форме которого разместите три полосы прокрутки для управления цветом (красным, зеленым, синим), который используется как фон формы. Начальные положения бегунков полос прокруток должны соответствовать заданному в дизайнера цвету фона формы

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 28: Создание проекта с использованием полос прокрутки для ввода информации

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать приложение, в котором бегунок и элемент управления NumericUpDown управляют индикатором прогресса. Дополнительное условие: бегунок и NumericUpDown должны работать синхронно. То есть, при изменении значения одного элемента, значение другого должно изменяться автоматически на ту же величину
2. Разработать проект, который позволяет пользователю вычислить факториал числа. Число, для которого рассчитывается факториал, выбирается с помощью элемента управления TrackBar. При щелчке по кнопке «Расчет», меняется надпись «Число n» на «N!» и в строке ввода выводится значение факториала числа

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 29: Создание проекта с использованием полос прокрутки для ввода информации

**Форма контроля:** Рубежный контроль

**Задание:**

1. Написать простое приложение, которое будет рисовать на главной форме 4000 квадратных клеток, каждая из которых будет пронумерована для удобства. Максимальное количество клеток по горизонтали будет 80, по вертикали - 50 штука
2. В приводимых ниже заданиях организовать вычисление с помощью полосы прокрутки для различных n. Причём, предусмотреть вычисление, как в

цикле, так и по формуле, приведенной в правой части выражения:  $1 + 3 + 5 + \dots + (2n - 1) = n^2$

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 30: Создание проекта с использованием группы зависимых переключателей

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создание формы с различными видами кнопок. Программа должна будет выполнять следующие функции: радиокнопки задают текст сообщения, которое будет выводиться по нажатию на обычную кнопку. Флажок должен определять — выводить сообщение или нет

2. Создайте Windows-приложение, на форме которого определите следующие элементы:

а) радиокнопки, которые управляют выбором изображения фона формы (4 варианты). При выборе радиокнопки сразу изменяется фон формы.

б) в комбинированном списке выбирается заголовок формы (4 варианта).

в) список с единичным выбором позволяет выбрать шрифт на всех элементах управления.

Во всех элементах должны быть выбраны значения по умолчанию, именно те, которые используются на форме при запуске приложения

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 31: Создание проекта с использованием группы зависимых переключателей

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создание формы с использованием списков. Необходимо написать программу, предназначенную для учета данных об участниках соревнований. Программа будет содержать два списка — `ComboBox`, для ввода информации об участниках, и `CheckedListBox`, для хранения и обработки данных. С помощью списка `ComboBox` пользователь будет выбирать фамилии лиц, которых необходимо добавить в список участников. Две кнопки на форме будут добавлять или удалять участников из списка

2. Создайте Windows-приложение, используя переключатели или флажки, для выбора цвета формы. При выборе цвета появляется сообщение с выбранным цветом.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 32: Создание проекта с использованием группы зависимых переключателей

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать Windows-приложение
2. Создайте форму, которая позволит ввести имя, адрес, род занятий и возраст. На форме должны располагаться кнопки ОК и Help. Данные из формы считываются и заносятся в результирующее текстовое поле, расположенное на этой же форме. При нажатии на кнопку Help выводится краткое описание каждого текстового окна в результирующем текстовом поле

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 33: Создание процедур на основе событий

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разработать приложение, с помощью которого можно вычислить время падения тела с некоторой высоты при условии, что высота может задаваться в метрах, сантиметрах и дюймах
2. Разработайте приложение, которое при выборе определенного цвета в группе переключателей, изменяет цвет формы

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 34: Создание процедур на основе событий

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создайте проект, для решения задачи с использованием классов: класс «Worker» поля: имя; возраст; вес (начальное значение 60).
2. Создайте новый проект ClassApp2 для решения следующей задачи с использованием классов

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 33-34. Разработка оконного приложения

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Как создать проект WinForms в Visual Studio?
2. Как работать с формой после создания проекта?
3. Как создать простое приложение с кнопкой и меткой, где при нажатии на кнопку текст метки изменится?

Практическая работа 35: Создание проекта с использованием кнопочных компонентов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Используя кнопочные компоненты button, разработать программу – калькулятор, выполняющий простейшие действия.
2. Составьте программу, которая переводит суммы из рублей в доллары
3. . Вычислить и вывести на экран таблицу значений функции  $y = a \cdot \ln(x)$  при  $x$ , изменяющемся от  $x_0$  до  $x_k$  с шагом  $dx$ ,  $a$  – константа

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 36: Создание проекта с использованием кнопочных компонентов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать программу для определения целочисленной матрицы  $15 \times 15$ . Разработать обработчик кнопки, который будет искать минимальный элемент на дополнительной диагонали матрицы. Результат вывести в текстовое поле.
2. Написать программу подсчета числа слов в произвольной строке. В качестве разделителя может быть любое число пробелов. Для ввода строк использовать ListBox. Строки вводятся на этапе проектирования формы, используя окно свойств. Вывод результата организовать в метку Label

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 37: Создание проекта с использованием кнопочных компонентов

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разместите на форме ряд кнопок (Button), и одно поле ввода (TextBox). Создайте обработчики события нажатия на данные кнопки, которые будут менять текст на нажатой кнопке. Текст на кнопке берется из поля ввода.
2. Разместите на форме две кнопки (Button) и одну метку (Label). Сделайте на кнопках следующие надписи: «скрыть», «показать». Создайте обработчики события нажатия на данные кнопки, которые будут скрывать или показывать метку. Создайте обработчик события создания формы (Load), который будет устанавливать цвет формы и менять текст метки на строку «Начало работы».

3. Разместите на форме поле ввода (TextBox), метку (Label) и кнопку (Button). Создайте обработчик события нажатия на кнопку, который будет копировать текст из поля ввода в метку. Создайте обработчик события нажатия кнопки мышки на форме (Click), который будет устанавливать цвет формы и менять текст метки на строку «Начало работы» и очищать поле ввода

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 38: Создание проекта с использованием компонентов стандартных диалогов и системы меню

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать главное меню
2. Создать вложенное меню
3. Написать обработчики сообщений меню
4. Создать контекстное меню
5. Добавить пометки пунктов меню

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 39: Создание проекта с использованием компонентов стандартных диалогов и системы меню

**Форма контроля:** Рубежный контроль

**Задание:**

1. Работа с редактором изображений
2. Добавить на форму панель инструментов
3. Добавить на форму строку состояния

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 40: Создание проекта с использованием компонентов стандартных диалогов и системы меню

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создайте Windows-приложение для изменения цвета фона формы через пункты меню (например, Красный, Синий, Белый) и кнопки панели инструментов. Название цвета фона должно выводиться в строке состояния
2. Создать программу «Тест»

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 41: Разработка функциональной схемы работы приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разработать программу, которая рассчитывает стоимость с учётом скидки. Пользователь вводит информацию о скидке, наименовании материалов, количестве и стоимости, а программа рассчитывает общую стоимость по каждому пункту, стоимость с учётом скидки и суммарный итог. Нарисовать блок-схему, отражающую логику работы приложения, используя любую онлайн-программу для создания диаграмм. В схеме должны быть обозначены основные модули: интерфейс пользователя, обработка данных, хранение информации. Обозначить потоки данных между модулями

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 42: Разработка функциональной схемы работы приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Доработайте функциональную схему системы комплексной автоматизации лечебно-профилактических учреждений (поликлиника)

2. Разработать функциональную схему работы приложения «Библиотека». Обязательные блоки: БД книг, БД читателей, Подсистема поиска, Справочная подсистема и т.д.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 43: Разработка функциональной схемы работы приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разработать функциональную схему работы приложения «Продажа бытовой техники». Обязательные блоки: БД товара, БД покупателей, Подсистема поиска, Поставщики товара и т.д. В схеме должны быть обозначены основные модули: интерфейс пользователя, обработка данных, хранение информации. Обозначить потоки данных между модулями

2. Разработать функциональную схему работы приложения «Автосервис». Обязательные блоки: БД услуг, БД клиентов (регистрационные данные транспорта), Склад запчастей и т.д. В схеме должны быть обозначены основные модули: интерфейс пользователя, обработка данных, хранение информации. Обозначить потоки данных между модулями

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 44: Разработка оконного приложения с несколькими формами

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать приложение с тремя формами: Главная, Опции и О программе. Форму Опции вызывать в обычном окне. Для вызова формы О программе использовать модальное окно. На рисунке показаны главная форма и подформы проекта

2. В зрительном зале 25 рядов, в каждом из которых 36 мест (кресел). Информация о проданных билетах хранится в двумерном массиве, номера строк которого соответствуют номерам рядов, а номера столбцов — номерам мест. Если билет на то или иное место продан, то соответствующий элемент массива имеет значение 1, в противном случае — 0 Составить программу, определяющую число проданных билетов на места в 12-м ряду

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 45: Разработка оконного приложения с несколькими формами

**Форма контроля:** Рубежный контроль

**Задание:**

1. Составить программу определения, в норме ли вес обследуемого пациента (нормой считается вес, равный  $(\text{рост(см)} - 100) \pm 5 \text{ кг}$ )

2. Создать приложение с несколькими формами

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 46: Разработка оконного приложения с несколькими формами

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создайте программу, в которой предусмотрена работа с несколькими окнами, организация парольного доступа к скрытым окнам и вывод сообщений об ошибках через MessageBox.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 47: Разработка игрового приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать игру «Подбери пару», в которой игрок должен подобрать пару скрытым значкам

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 48: Разработка игрового приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Создать игру «Змейка»

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Практическая работа 49: Разработка игрового приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Разработайте любую игру на выбор

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

Теоретическое занятие 35. Этапы разработки приложения

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Какие типы приложений можно разрабатывать в Microsoft Visual Studio?
2. Какие этапы входят в разработку структуры приложения?
3. Что такое интерфейс пользователя?
4. Назовите принципы разработки интерфейса пользователя?

Теоретическое занятие 36. Этапы разработки приложения

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов



### **Вопросы:**

1. Что такое тестирование приложения?
2. Какие существуют типы тестирования программы?
3. Какие типы ошибок в программах могут встречаться?
4. Что такое отладка приложения?
5. Назовите распространенные методы отладки.

### Практическая работа 50: Разработка многооконного приложения.

Компиляция и запуск приложения

**Форма контроля:** Рубежный контроль

#### **Задание:**

1. При помощи компонентов ComboBox, ListBox, RadioButton, NumericUpDown выводить изображения через pictureBox. Компоненты ComboBox, ListBox, RadioButton, NumericUpDown будут разделены через компонент TabControl, содержащий несколько вкладок. На каждой вкладке будет свой компонент pictureBox и один из компонентов управления. Для удобства перехода между вкладками будут созданы кнопки перехода на следующую и/или предыдущую вкладку, а на последней вкладке будет кнопка закрытия программы.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 51: Разработка многооконного приложения.

Компиляция и запуск приложения

**Форма контроля:** Рубежный контроль

#### **Задание:**

1. Создать проект с тремя формами. Первая, она же главная, будет иметь две кнопки открыть подчиненное окно и открыть диалоговое окно, вторая форма будет считать, сколько раз её уже открыли, а при попытке повторно открыть её с главной формы, будет появляться сообщение о закрытии формы. Третья форма (диалоговое окно) будет изменять заголовки главного окна и подчинённого окна.

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 52: Разработка многооконного приложения.

Компиляция и запуск приложения

**Форма контроля:** Рубежный контроль

#### **Задание:**

1. Создать простую программу для просмотра изображений

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 53: Разработка интерфейса пользователя

**Форма контроля:** Рубежный контроль

**Задание:**

1. Необходимо построить приложение, содержащее форму входа пользователя в систему и форму регистрации нового пользователя. Необходимо также написать процедуры обработки событий для кнопок Регистрация (вызов формы Регистрация нового пользователя), Сброс (очистка окна), Выход (выход из приложения), Заккрыть (Заккрыть форму)

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 54: Разработка интерфейса пользователя

**Форма контроля:** Рубежный контроль

**Задание:**

1. В соответствии с вариантом задания разработать интерфейс основных программных модулей с использованием WindowsForms

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 55: Разработка интерфейса пользователя

**Форма контроля:** Рубежный контроль

**Задание:**

1. В соответствии с вариантом задания разработать интерфейс основных программных модулей с использованием WindowsForms

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 56: Тестирование, отладка приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Протестируйте созданное приложение, проверив весь функционал программы, вводя разные входные данные. В случае нахождения ошибок, исправьте их. Заполните таблицу

2. Протестируйте созданное приложение в Практической работе № 55, проверив весь функционал программы, вводя разные входные данные. В случае нахождения ошибок, исправьте их

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Практическая работа 57: Тестирование, отладка приложения

**Форма контроля:** Рубежный контроль

**Задание:**

1. Протестируйте созданное приложение в Практической работе № 53, проверив весь функционал программы, вводя разные входные данные

2. Протестируйте созданное приложение в Практической работе № 54, проверив весь функционал программы, вводя разные входные данные

Порядок выполнения работы согласно методическим рекомендациям по практическим работам по ОП.04 «Основы алгоритмизации и программирования»

### Самостоятельная работа 3. Интерфейс пользователя

**Форма контроля:** Рубежный контроль

**Задание:**

Подготовить презентацию по данной теме

Порядок выполнения работы согласно методическим рекомендациям по самостоятельным работам по ОП.04 «Основы алгоритмизации и программирования».

### Теоретическое занятие 37. Иерархия классов

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое класс в контексте ООП?
2. Что такое иерархия классов и как она позволяет определять новые классы на основе уже имеющихся?
3. Как наследование классов помогает упорядочивать и ранжировать классы, объединять общие для нескольких классов свойства в одном классе и использовать его в качестве базового?
4. Какие особенности наследования классов?

### Теоретическое занятие 38. Иерархия классов

**Форма контроля:** оперативный контроль.

**Задание:** ответить на вопросы устно (фронтальный опрос).

**Условия выполнения задания:**

- 1) обучающиеся устно отвечают на вопросы;
- 2) время, отводимое на опрос – 10 мин;
- 3) максимальный балл за задание – 5 баллов

**Вопросы:**

1. Что такое перегрузка методов?
2. Как работает перегрузка методов?

3. Какие принципы нужно учитывать при перегрузке методов?
4. Какие преимущества дает перегрузка методов?

### 3. Критерии оценки

#### 3.1. Инвариантные критерии оценки

#### Критерии оценки устных (письменных) ответов на теоретические вопросы

Критерии оценки		Оценка
<b>1</b>	Демонстрирует глубокое, полное знание и понимание программного материала. Последовательно, самостоятельно раскрывает основное содержание вопроса. Выводы аргументированы, основаны на самостоятельно выполненном анализе, обобщении данных. Четко и верно даны определения понятий и научных терминов. Дает верные, самостоятельные ответы на вопросы.	5 (отлично)
<b>2</b>	Демонстрирует недостаточно глубокое, полное знание и понимание программного материала. Недостаточно последовательно, но самостоятельно раскрывает основное содержание вопроса. Выводы основаны на самостоятельно выполненном анализе, обобщении данных, но в отдельных случаях недостаточно аргументированы. Недостаточно четко и верно даны определения понятий и научных терминов. При ответе на вопросы допускает несущественные ошибки, которые может исправить самостоятельно.	4 (хорошо)
<b>3</b>	Демонстрирует в отдельных вопросах, неглубокое владение знаниями программного материала. Излагает программный материал фрагментарно, не всегда последовательно. Допущены ошибки и неточности в использовании научной терминологии. При ответе на вопросы допускает неточности.	3 (удовлетворительно)
<b>4</b>	Студент демонстрирует незнание и непонимание программного материала. Основное содержание учебного материала не раскрыто; допущены грубые ошибки в определении понятий, при использовании терминологии. Затрудняется отвечать на вопросы, при ответе допускает серьезные ошибки.	2 (неудовлетворительно)

### **Критерии оценки результатов выполнения тестового задания**

<b>Оценка</b>	<b>Количество правильных ответов на вопросы в % соотношении от общего числа вопросов</b>
Оценка 5 «отлично»	90-100%
Оценка 4 «хорошо»	76-89%
Оценка 3 «удовлетворительно»	50-75%
Оценка 2 «неудовлетворительно»	≤ 49%

### **Критерии оценки результатов промежуточной аттестации:**

<b>Критерии оценки</b>		<b>Оценка</b>
<b>1</b>	обучающийся показал полный объем знаний по вопросу, владеет культурой общения, навыками научного изложения материала, устанавливает связь между теоретическими знаниями и способами практической деятельности; ясно, точно и логично отвечает на заданные вопросы.	5 (отлично)
<b>2</b>	обучающийся логично и научно изложил материал, но недостаточно полно определяет практическую значимость теоретических знаний; не высказывает своей точки зрения по данному вопросу, не смог дать достаточно полного ответа на поставленные вопросы	4 (хорошо)
<b>3</b>	обучающийся при раскрытии вопроса допустил содержательные ошибки, испытывает затруднения при ответе на большинство вопросов.	3 (удовлетворительно)
<b>4</b>	обучающийся показал слабые теоретические и практические знания, допустил грубые ошибки при раскрытии вопроса, не смог ответить на заданные вопросы	2 (неудовлетворительно)

## РУБЕЖНЫЙ КОНТРОЛЬ (ТЕСТИРОВАНИЕ)

1. Что такое язык программирования

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) система, определяющая воздействие одного объекта на другой
- Б) формальная знаковая система, предназначенная для описания алгоритма в форме
- В) выделение существенных характеристик некоторого объекта

2. Что определяет язык программирования?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) организацию соответствующих условий;
- Б) последовательность двоичных кодов
- В) набор лексических, синтаксических и семантических правил

3. На какие группы можно разделить множество языков программирования

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) язык ассемблера
- Б) языки низкого и высокого уровней
- В) язык мнемокодов

4. Что составляется из символов?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) множество символов, используемых на языке
- Б) ключевые слова языка
- В) двоичные коды

5. Что называется трансляцией?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) процесс обработки программы
- Б) процесс интерпретации
- В) процесс перевода

6. Что такое компилятор?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) программа, преобразующая исходный код в машинный код
- Б) программа для выполнения кода построчно
- В) текстовый редактор для программистов

7. Что такое парадигма программирования?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) совокупность идей и понятий о способах построения программ
- Б) набор команд конкретного языка
- В) система типов данных

8. Что такое интерпретатор?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

А) программа для перевода кода в машинные инструкции по строкам

Б) компилятор

В) транслятор

9. Что такое синтаксис языка программирования?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

А) набор команд

Б) правила написания кода

В) структура программы

10. Что такое алгоритм?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

а) набор команд

б) программа

в) последовательность действий для решения задачи

**Критерии оценивания:**

Отметка	Объем выполнения работы в %
«5» (отлично)	90 – 100
«4» (хорошо)	70 – 89
«3» (удовлетворительно)	50 – 69
«2» (неудовлетворительно)	Менее 50

**Ключ ответов к тестовым заданиям**

№ вопроса	Правильный ответ	Проверяемые компетенции
1	Б	ОК 01, ОК 02, ОК 05, ОК 09
2	В	ОК 01, ОК 02, ОК 05, ОК 09
3	Б	ОК 01, ОК 02, ОК 05, ОК 09
4	Б	ОК 01, ОК 02, ОК 05, ОК 09
5	В	ОК 01, ОК 02, ОК 05, ОК 09
6	А	ОК 01, ОК 02, ОК 05, ОК 09
7	А	ОК 01, ОК 02, ОК 05, ОК 09
8	А	ОК 01, ОК 02, ОК 05, ОК 09
9	Б	ОК 01, ОК 02, ОК 05, ОК 09
10	В	ОК 01, ОК 02, ОК 05, ОК 09

## МЕЖСЕССИОННЫЙ КОНТРОЛЬ (ТЕСТИРОВАНИЕ)

1. Какие циклы существуют в языке C#

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) for, while
- Б) for, while, do while, foreach
- В) for, while, do while

2. Какие типы переменных существуют

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) int, char, bool, float, double
- Б) numeric, decimal, int
- В) string, decimal, bool, integer

3. Что такое константа

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) глобальная переменная
- Б) переменная которая может быть изменена в любое время
- В) переменная значение которой нельзя изменить

4. Какая переменная правильно объявлена

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) int x=5;
- Б) x=0;
- В) \$x=10;

5. Что такое цикл и для чего они нужны

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) циклы нужны для многократного выполнения кода
- Б) циклы нужны для многократного запуска программы
- В) циклы нужны для многократного размещения данных

6. Какие бывают массивы

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) одномерные
- Б) многомерные
- В) оба варианты верные

7. Что такое массив

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) набор однотипных данных, которые располагаются в памяти последовательно друг за другом
- Б) набор данных типа int (32-бит целое)
- В) набор текстовых значений в формате Unicode, которые расположены в случайном порядке



8. Какой тип переменной используется в коде: `int a = 5`

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) знаковое 64-бит целое
- Б) знаковое 8-бит целое
- В) знаковое 32-бит целое

9. Что делает оператор «%»

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) возвращает процент от суммы
- Б) возвращает остаток от деления
- В) возвращает тригонометрическую функцию

10. Что сделает программа, выполнив следующий код: `Console.WriteLine(«Hello, World!»);`

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) напишет Hello, World!
- Б) напишет на новой строке Hello, World!
- В) удалит все значения с Hello, World!

11. Для чего нужны условные операторы

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) чтобы устанавливать условия пользователю
- Б) для оптимизации программы
- В) для ветвления программы

12. Как сделать инкрементацию числа

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) `!=`
- Б) `++`
- В) `--`

13. Чему равен `d`, если `int a = 0; int b = a++; int c = 0; int d = a + b + c + 3;`

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) 4
- Б) 3
- В) false

14. Как найти квадратный корень из числа `x`

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) `Sqrt(x)`
- Б) `Math.Sqrt(x)`
- В) `Arifmetic.sqrt(x)`

15. Обозначения оператора «НЕ»

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- A) No  
 Б) !  
 В) Not

**Критерии оценивания:**

Отметка	Объем выполнения работы в %
«5» (отлично)	90 – 100
«4» (хорошо)	70 – 89
«3» (удовлетворительно)	50 – 69
«2» (неудовлетворительно)	Менее 50

**Ключ ответов к тестовым заданиям**

№ вопроса	Правильный ответ	Проверяемые компетенции
1	Б	ОК 01, ОК 02, ОК 05, ОК 09
2	А	ОК 01, ОК 02, ОК 05, ОК 09
3	В	ОК 01, ОК 02, ОК 05, ОК 09
4	А	ОК 01, ОК 02, ОК 05, ОК 09
5	А	ОК 01, ОК 02, ОК 05, ОК 09
6	В	ОК 01, ОК 02, ОК 05, ОК 09
7	А	ОК 01, ОК 02, ОК 05, ОК 09
8	В	ОК 01, ОК 02, ОК 05, ОК 09
9	Б	ОК 01, ОК 02, ОК 05, ОК 09
10	Б	ОК 01, ОК 02, ОК 05, ОК 09
11	В	ОК 01, ОК 02, ОК 05, ОК 09
12	Б	ОК 01, ОК 02, ОК 05, ОК 09
13	А	ОК 01, ОК 02, ОК 05, ОК 09
14	Б	ОК 01, ОК 02, ОК 05, ОК 09
15	Б	ОК 01, ОК 02, ОК 05, ОК 09

## ИТОГОВЫЙ КОНТРОЛЬ (ТЕСТИРОВАНИЕ)

1. Что такое язык программирования

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) система, определяющая воздействие одного объекта на другой
- Б) формальная знаковая система, предназначенная для описания алгоритма в форме
- В) выделение существенных характеристик некоторого объекта

2. Как сделать декрементацию числа

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) !=
- Б) --
- В) %

3. Свойство, при котором объекты содержат описание атрибутов и действий одновременно

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) наследование
- Б) полиморфизм
- В) инкапсуляция

4. Чему будет равен с, если `int a = 0; int c = -a;`

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) null
- Б) 1
- В) -1

5. Обозначение оператора «ИЛИ»

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) @@
- Б) &&
- В) ||

6. Один из принципов объектно-ориентированного программирования

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) наследование
- б) отдача
- в) передача

7. Совокупность объектов, характеризующаяся общностью методов и свойств

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) класс
- б) группа
- в) вид

8. Под объектами понимают

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) всю абстрактную сущность, заданную набором имен атрибутов и имен методов поведения
- б) некоторую абстрактную сущность, заданную набором имен атрибутов и имен методов поведения
- в) некоторую видимую сущность, заданную набором имен атрибутов и имен методов поведения

9. Обозначение оператора «И»

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) &&
- б) and
- в) ||

10. Свойство объектов, при котором действие с одинаковыми именами вызывает различное поведение для различных объектов

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) полиморфизм
- б) отдача
- в) монорфизм

11. Изменение состояния объекта в ответ на какое-либо действие

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) метод
- б) событие
- в) значение

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

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) абстрактными
- б) видимыми
- в) визуальными

13. Свойство Name отвечает за

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) название компонента
- б) имя элемента
- в) назначение элемента

14. Этот компонент предназначен для вывода текста на поверхность формы

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) Label

- б) TextBox
- в) EditText

15. Когда необходимо составлять блок-схему программы

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) после составления программы
- б) в процессе составления программы
- в) до начала составления самой программы

16. Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) компилятор
- б) ассемблер
- в) интерпретатор

17. Как называется алгоритм, в котором действия выполняются друг за другом, не повторяясь

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- а) циклическим
- б) линейным
- в) разветвленным

18. Символьный тип данных объявляется служебным словом

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) string
- Б) bool
- В) char

19. Цикл с предусловием определяется служебным словом

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) for
- Б) while
- В) do while

20. Определите, если число повторений цикла известно и задано наибольшее допустимое значение n, то лучше использовать

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) цикл с предусловием
- Б) цикл с параметром
- В) цикл с постусловием

21. Как называется набор однотипных данных, имеющий общее для всех своих элементов имя?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) массив
- Б) множество
- В) запись

22. На какие группы можно разделить множество языков программирования

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) язык ассемблера
- Б) языки низкого и высокого уровней
- В) язык мнемочкодов

23. Что такое синтаксис языка программирования?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) набор команд
- Б) правила написания кода
- В) структура программы

24. Что такое тестирование программы?

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) процесс проверки кода на наличие ошибки
- Б) процесс поиска и исправления ошибок в программе
- В) нет правильного ответа

25. Что сделает программа, выполнив следующий код: Console.WriteLine(«Hello, World!»);

**Проверяемые компетенции:** ОК 01, ОК 02, ОК 05, ОК 09

- А) напишет Hello, World!
- Б) напишет на новой строчке Hello, World!
- В) удалит все значения с Hello, World!

#### Критерии оценивания:

Отметка	Объем выполнения работы в %
«5» (отлично)	90 – 100
«4» (хорошо)	70 – 89
«3» (удовлетворительно)	50 – 69
«2» (неудовлетворительно)	Менее 50

#### Ключ ответов к тестовым заданиям

№ вопроса	Правильный ответ	Проверяемые компетенции
1	Б	ОК 01, ОК 02, ОК 05, ОК 09
2	Б	ОК 01, ОК 02, ОК 05, ОК 09
3	В	ОК 01, ОК 02, ОК 05, ОК 09
4	В	ОК 01, ОК 02, ОК 05, ОК 09
5	В	ОК 01, ОК 02, ОК 05, ОК 09

6	A	OK 01, OK 02, OK 05, OK 09
7	A	OK 01, OK 02, OK 05, OK 09
8	Б	OK 01, OK 02, OK 05, OK 09
9	A	OK 01, OK 02, OK 05, OK 09
10	A	OK 01, OK 02, OK 05, OK 09
11	Б	OK 01, OK 02, OK 05, OK 09
12	В	OK 01, OK 02, OK 05, OK 09
13	Б	OK 01, OK 02, OK 05, OK 09
14	A	OK 01, OK 02, OK 05, OK 09
15	В	OK 01, OK 02, OK 05, OK 09
16	A	OK 01, OK 02, OK 05, OK 09
17	Б	OK 01, OK 02, OK 05, OK 09
18	В	OK 01, OK 02, OK 05, OK 09
19	Б	OK 01, OK 02, OK 05, OK 09
20	В	OK 01, OK 02, OK 05, OK 09
21	A	OK 01, OK 02, OK 05, OK 09
22	Б	OK 01, OK 02, OK 05, OK 09
23	Б	OK 01, OK 02, OK 05, OK 09
24	A	OK 01, OK 02, OK 05, OK 09
25	Б	OK 01, OK 02, OK 05, OK 09

**Вопросы для экзамена по ОП.04 «Основы алгоритмизации и программирования»**

**Теоретические вопросы:**

1. Языки программирования
2. Жизненный цикл программы. Программа. Программный продукт и его характеристики
3. Основные этапы решения задач на компьютере
4. Типы данных
5. Условный оператор. Оператор выбора
6. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы
7. Массивы. Одномерные и двумерные массивы
8. Строки. Стандартные процедуры и функции для работы со строками
9. Структурированный тип данных – множество. Операции над множествами
10. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа. Организация доступа к файлам
11. Процедуры и функции
12. Рекурсивные функции
13. Структурное программирование
14. Модульное программирование
15. Указатели
16. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс
17. Основные принципы ООП: инкапсуляция, наследование, полиморфизм
18. Классы объектов. Компоненты и их свойства
19. Интегрированная среда разработчика Visual Studio
20. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение
21. Дополнительные элементы управления. Свойства компонентов
22. События компонентов, их сущность и назначение. Создание процедур на основе событий
23. Разработка оконного приложения
24. Создание интерфейса приложения
25. Тестирование и отладка приложения
26. Классы ООП: виды, назначения, свойства, методы, события
27. Перегрузка методов

**Практические задания:**

1. Составление программ линейной структуры
2. Составление программ разветвляющей структуры
3. Составление программ циклической структуры
4. Обработка одномерных массивов
5. Обработка двумерных массивов
6. Работа со строками



7. Работа с данными типа множество
8. Составление программ с использованием текстовых файлов
9. Организация процедур
10. Организация функций
11. Применение рекурсивных функций
12. Создание проекта с использованием компонентов для работы с текстом
13. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени
14. Создание проекта с использованием полос прокрутки для ввода информации
15. Создание проекта с использованием группы зависимых переключателей
16. Создание процедур на основе событий
17. Создание проекта с использованием кнопочных компонентов
18. Создание проекта с использованием компонентов стандартных диалогов и системы меню
19. Разработка оконного приложения с несколькими формами
20. Разработка многооконного приложения
21. Разработка интерфейса приложения
22. Тестирование, отладка приложения