Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный индустриальный университет»

Кафедра прикладных информационных технологий и программирования

УТВЕРЖДАЮ
Директор института
информационных технологий и
автоматизированных систем
_____ А.В. Зимин
подпись
«____» _____ 20___ г.

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Современные технологии программирования

(* Перечень направлений подготовки (специальностей) и направленностей (профилей) на следующей странице)

Форма обучения
Очная форма

Год начала подготовки 2025

Новокузнецк
2025

СибГИУ
Подлинник электронного документа хранится в системе электронного согласования документов СибГИУ
ДОКУМЕНТ ПОДПИСАН ЭП
Сертификат: daba56d7b892efca3bf5c04ed0251c98b19b099c
Владелец: Директор ИИТиАС Зимин Алексей Валерьевич
Действителен: с 15 января 2025 г. по 10 апреля 2026 г.
Дата и время: 30-04-2025 09:50:56 ПРОВЕРИТЬ ПОДПИСЬ

Перечень направлений подготовки (специальностей) и направленностей (профилей):

09.03.01 «Информатика и вычислительная техника»
(направленность (профиль): «Программная инженерия»)
Квалификация выпускника: «Бакалавр»
Срок обучения: 3 года 5 месяцев

09.03.01 «Информатика и вычислительная техника»
(направленность (профиль): «Программная инженерия»)
Квалификация выпускника: «Бакалавр»
Срок обучения: 4 года

## 1 Цели и задачи освоения учебной дисциплины

Целями учебной дисциплины являются:
– получение теоретических сведений о современных технологиях в объектно-ориентированном программировании;
– наработка практических навыков использования современных технологий программирования.

Задачами учебной дисциплины являются:
– изучение современных конструкций программирования с использованием классов на примере языка C#;
– изучение ряда элементов иерархической структуры классов .Net Framework.

## 2 Место учебной дисциплины в структуре ООП по направлению подготовки (специальности)

Учебная дисциплина относится к учебным дисциплинам части, формируемой участниками образовательных отношений **Блока 1 «Дисциплины (модули)»** ООП по направлению подготовки (специальности) 09.03.01 «Информатика и вычислительная техника».

Учебная дисциплина базируется на предварительном усвоении обучающимися учебных дисциплин:
– Основы программирования;
– Программирование.

Учебная дисциплина дополняет знания и умения, получаемые по одновременно изучаемым и последующим учебным дисциплинам:
– Выполнение и защита выпускной квалификационной работы.

## 3 Планируемые результаты обучения по учебной дисциплине

Процесс изучения учебной дисциплины направлен на формирование следующих компетенций:

### – Профессиональные компетенции

| Наименование категории (группы) ПК | Код и наименование ПК | Код и наименование индикатора достижения ПК | Планируемые результаты обучения |
|---|---|---|---|
| | ПК-1: Способен участвовать в анализе возможностей реализации требований к компьютерному программному обеспечению | ПК-1.1 Собирает, систематизирует, выявляет связи и документирует требования к компьютерному программному обеспечению | – знать: виды требований к программному обеспечению.<br>– уметь: выявлять связи между элементами программного обеспечения. |
| | | ПК-1.2 Участвует в оценке времени и | – знать: типовые архитектуры |

| | | | |
|---|---|---|---|
| | | трудоемкости реализации требований к компьютерному программному обеспечению | программного обеспечения и трудоемкость их реализации.<br>– уметь: реализовывать типовые архитектуры за требуемое время. |
| | | ПК-1.3 Участвует в согласовании требований к компьютерному программному обеспечению с заинтересованными сторонами | – знать: перечень возможных заинтересованных сторон.<br>– уметь: выявлять требования заинтересованных сторон. |
| | ПК-2: Способен участвовать в проектировании компьютерного программного обеспечения | ПК-2.3 Участвует в разработке технической документации на программное обеспечение с использованием существующих стандартов | – знать: существующие стандарты и типовые инструкции, применяемые для разработки технической документации.<br>– уметь: применять существующие стандарты для разработки технической документации на компьютерное программное обеспечение. |

## 4 Объем и содержание учебной дисциплины

Учебные занятия по учебной дисциплине проводятся в форме контактной работы и в форме самостоятельной работы обучающихся.

Контактная работа включает в себя занятия лекционного типа (лекции), занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы), промежуточную аттестацию обучающихся и иные формы взаимодействия обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации ООП на иных условиях, в том числе при проведении промежуточной аттестации обучающихся. Контактная работа может проводиться с применением электронного обучения, дистанционных образовательных технологий.

### Объем учебной дисциплины

| Семестр / курс | ***ИТОГО*** | ***6 семестр*** |
|---|---|---|
| Форма промежуточной аттестации | | *экзамен, зачет с оценкой по КР* |

| | | | |
|---|---|---|---|
| Трудоёмкость | *академ. час.* | ***216*** | *216* |
| | *зачетных единиц* | ***6*** | *6* |
| Лекции, *академ. час.* | | ***32*** | *32* |
| в форме практической подготовки | | ***0*** | *0* |
| Лабораторные работы, *академ. час.* | | ***16*** | *16* |
| в форме практической подготовки | | ***0*** | *0* |
| Практические занятия, *академ. час.* | | ***32*** | *32* |
| в форме практической подготовки | | ***0*** | *0* |
| Курсовая работа, *академ. час.* | | ***36*** | *36* |
| в форме практической подготовки | | ***0*** | *0* |
| Консультации, *академ. час.* | | ***0*** | *0* |
| в форме практической подготовки | | ***0*** | *0* |
| Самостоятельная работа, *академ. час.* | | ***82*** | *82* |
| в форме практической подготовки | | ***0*** | *0* |
| Контроль, *академ. час.* | | ***18*** | *18* |
| в форме практической подготовки | | ***0*** | *0* |

## Содержание учебной дисциплины

Раздел 1 Некоторые технологии в программировании;

Тема 1.1 Основные элементы класса (Практическое занятие, предусматривающее создание программы с использованием собственного класса, имеющего все основные структурные элементы);

Тема 1.2 Перегрузка операторов (Общие правила перегрузки для унарных и бинарных операторов. Перегрузка операторов сложения/вычитания, инкремента/декремента, проверки на равенство/неравенство);

Тема 1.3 Делегаты (Понятие делегата. Формат описания. Групповые операции с делегатами. Ковариантность и контравариантность);

Тема 1.4 Анонимные методы и лямбда-выражения (Понятие анонимной функции. Реализация анонимной функции с использованием анонимного метода. Реализация анонимной функции с использованием лямбда-выражения);

Тема 1.5 События (Понятие события, общая схема его описания. Создание обработчика событий. Формат событий в среде .NET Framework);

Раздел 2 Универсальные типы в C#;

Тема 2.1 Общая схема построения универсальных типов (Понятие и назначение универсальных типов. Общая форма описания универсальных типов);

Тема 2.2 Ограничения, накладываемые в универсальных типах (Наложение ограничений на параметры типа: на базовый класс, на интерфейс, на конструктор, на глобальный тип параметра. Задание связей между параметрами типа);

Тема 2.3 Параметры типы в методах (Использование параметров типа в методах);

Тема 2.4 Некоторые типы библиотеки .Net Framework (Рассмотрение основных свойств, методов и событий классов: массив, списки, очереди и стеки, множества, словари);

Раздел 3 Язык интегрированных запросов LINQ;

Тема 3.1 Введение в LINQ (Понятие LINQ. Программные конструкции и типы, используемые в LINQ: методы расширения, анонимные типы, интерфейс перечислителя);

Тема 3.2 Построение запросов на LINQ (Общая структура запроса. Использование конструкций запроса: отбор, условия, сортировки, группировки, запросы с несколькими источниками данных);

Тема 3.3 Использование методов интерфейса перечислителя (Сопоставление между конструкциями запроса и методами интерфейса перечислителя. Рассмотрение некоторых «дополнительных» методов. Совместное использование двух форм запроса);

Раздел 4 Регулярные выражения.

## 5 Перечень тем лекций

| № раздела / темы дисциплины | Темы лекций | Трудоемкость, *академ.час* | |
|---|---|---|---|
| | | **всего** | **в форме практической подготовки** |
| Тема 1.2. | Перегрузка операторов | 2 | |
| Тема 1.3. | Делегаты | 1 | |
| Тема 1.4. | Анонимные методы и лямбда-выражения | 2 | |
| Тема 1.5. | События | 1 | |
| Тема 2.1. | Общая схема построения универсальных типов | 2 | |
| Тема 2.2. | Ограничения, накладываемые в универсальных типах | 2 | |
| Тема 2.3. | Параметры типы в методах | 2 | |
| Тема 2.4. | Некоторые типы библиотеки .Net Framework | 6 | |
| Тема 3.1. | Введение в LINQ | 2 | |
| Тема 3.2. | Построение запросов на LINQ | 8 | |
| Тема 3.3. | Использование методов интерфейса перечислителя | 2 | |
| Раздел 4. | Регулярные выражения | 2 | |
| **Итого:** | | **32** | **0** |

## 6 Перечень тем практических занятий (семинаров)

| № раздела / темы дисциплины | Темы практических занятий (семинаров) | Трудоемкость, *академ.час* | |
|---|---|---|---|
| | | **всего** | **в форме** |

| | | | практической подготовки |
|---|---|---|---|
| Тема 1.1. | Основные элементы класса | 6 | |
| Тема 1.2. | Перегрузка операторов | 6 | |
| Тема 1.3. | Использование делегатов | 6 | |
| Тема 1.4;<br>Тема 2.4. | Универсальные типы библиотеки .Net Framework | 8 | |
| Тема 1.5. | События | 6 | |
| **Итого:** | | **32** | **0** |

## 7 Перечень тем лабораторных работ

| № раздела / темы дисциплины | Темы лабораторных работ | Трудоемкость, *академ.час* | |
|---|---|---|---|
| | | всего | в форме практической подготовки |
| Тема 3.2;<br>Тема 3.3. | Работа средствами LINQ с одним источником данных | 6 | |
| Тема 3.2;<br>Тема 3.3. | Работа средствами LINQ с несколькими источниками данных | 6 | |
| Раздел 4. | Регулярные выражения | 4 | |
| **Итого:** | | **16** | **0** |

## 8 Перечень тем курсовых работ (проектов)

| № раздела / темы дисциплины | Темы курсовых работ (проектов) | Трудоемкость, *академ.час* | |
|---|---|---|---|
| | | всего | в форме практической подготовки |
| Раздел 1;<br>Раздел 2;<br>Раздел 3. | Разработка и реализация прикладного программного обеспечения с использованием современных технологий программирования | 36 | |
| **Итого:** | | **36** | **0** |

## 9 Виды самостоятельной работы

| № раздела / темы дисциплины | Виды самостоятельной работы | Трудоемкость, *академ.час* | |
|---|---|---|---|
| | | всего | в форме практической подготовки |
| Раздел 1. | 1. Изучение лекционного материала;<br>2. Изучение теоретического | 28 | |

| | | | |
|---|---|---|---|
| | материала;<br>3. Подготовка к практическому занятию;<br>4. Прохождение тестирования. | | |
| Раздел 2. | 1. Изучение лекционного материала;<br>2. Изучение теоретического материала;<br>3. Подготовка к практическому занятию;<br>4. Прохождение тестирования. | 10 | |
| Раздел 3. | 1. Изучение лекционного материала;<br>2. Изучение теоретического материала;<br>3. Подготовка к лабораторной работе;<br>4. Прохождение тестирования. | 28 | |
| Раздел 4. | 1. Изучение лекционного материала;<br>2. Изучение теоретического материала;<br>3. Подготовка к лабораторной работе;<br>4. Прохождение тестирования. | 16 | |
| *Курсовая работа* | *Выполнение курсовой работы* | 36 | 0 |
| *Контроль* | *Подготовка к экзамену* | 18 | |
| **Итого:** | | **136** | **0** |

## 10 Матрица компетентностных задач по учебной дисциплине

| Контролируемые разделы (темы) дисциплины | Контролируемые компетенции | Оценочные средства |
|---|---|---|
| Раздел 1. | ПК-1;<br>ПК-2 | 1. Наблюдение за работой на практическом занятии;<br>2. Оценивание этапов выполнения курсовой работы;<br>3. Тестовые задания |
| Раздел 2. | ПК-1;<br>ПК-2 | 1. Наблюдение за работой на практическом занятии;<br>2. Оценивание этапов выполнения курсовой работы;<br>3. Тестовые задания |

| | | |
|---|---|---|
| Раздел 3. | ПК-1;<br>ПК-2 | 1. Наблюдение за работой на лабораторной работе;<br>2. Оценивание этапов выполнения курсовой работы;<br>3. Тестовые задания |
| Раздел 4. | ПК-1;<br>ПК-2 | 1. Наблюдение за работой на лабораторной работе;<br>2. Тестовые задания |
| Курсовая работа. | ПК-1;<br>ПК-2 | 1. Вопросы к защите курсовой работы |
| Экзамен. | ПК-1;<br>ПК-2 | 1. Экзамен на основе результатов компьютерного тестирования |

## 11 Входной контроль

**11.1 Цель входного контроля** − определить начальный уровень подготовленности обучающихся и выстроить индивидуальную траекторию обучения. В условиях личностно-ориентированной образовательной среды результаты входного оценивания обучающегося используются как начальные значения в индивидуальном профиле академической успешности обучающегося.

### 11.2 Описание оценочных средств

Форма проведения входного контроля – решение практической задачи.

Длительность – 30 минут. Проводится во время первого практического занятия.

**Шкала оценивания результатов входного контроля**

| Критерий оценивания | Оценка |
|---|---|
| задача решена правильно | зачтено |
| решение задачи дает неправильный ответ | не зачтено |

Основываясь на результатах контроля, формируется индивидуальная траектория обучающегося (группы обучающихся), исходя из следующих соображений:

– зачтено – обучающийся готов к изучению учебной дисциплины без дополнительного повторения изученного ранее материала;

– не зачтено – обучающийся имеет слабую подготовку к изучению учебной дисциплины, необходимы корректирующие занятия и консультации.

**Тестовые задания для входного контроля знаний обучающихся приведены в приложении.**

## 12 Текущий контроль

**12.1 Текущий контроль знаний** используется для оперативного и регулярного управления учебной деятельностью (в том числе самостоятельной) обучающихся. Данный вид контроля стимулирует у обучающихся стремление к систематической самостоятельной работе по изучению учебной дисциплины и позволяет оценить степень освоения изученного материала.

Текущий контроль успеваемости осуществляется в течение семестра, в ходе повседневной учебной работы по индивидуальной инициативе педагогического работника, в том числе посредством использования СУО «Moodle».

Формы проведения текущего контроля – наблюдение за работой на лабораторной работе; наблюдение за работой на практическом занятии; оценивание этапов выполнения курсовой работы; тестовые задания.

**12.2 Описание оценочных средств**

**Шкалы оценивания (методики оценки)**

**12.2.1 Оценивание этапов выполнения курсовой работы**

Курсовая работа – конечный продукт, получаемый в результате планирования и выполнения комплекса учебных и исследовательских заданий. Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве, а также уровень сформированности аналитических, исследовательских умений, умений практического и творческого мышления.

Оценивание этапов выполнения курсовой работы проводится в рамках текущей аттестации (три раза в семестр) в соответствии с графиком выполнения курсовой работы по учебной дисциплине. Результаты аттестации обучающихся по выполнению ими курсовой работы оцениваются руководителями в процентах от общего объема выполненной работы. Сроки выполнения курсовой работы устанавливаются кафедрой в графике выполнения курсовой работы. График выполнения курсовой работы выдается старостам групп в начале каждого семестра.

Критерии оценки % выполнения курсовой работы

| Этап работы | % выполнения | Форма отчетности |
|---|---|---|
| 1. Теоретический анализ задачи: введение, постановка задачи, выделение классов и событий, построение схемы функционирования моделируемой задачи | 30 | схема функционирования |

| | | |
|---|---|---|
| 2. Разработка структуры классов, реализация классов и событий | 60 | программная реализация классов |
| 3. Полная реализация программы | 100 | программная реализация задания |

Критерии оценки курсовой работы

| Критерии оценки выполнения курсовой работы | Оценка |
|---|---|
| Полное верное решение, задача решена рациональным способом, пояснительная записка имеет необходимый объем, структуру и правильность оформления, во время защиты работы обучаемый ответил на все вопросы относительно своего решения. | отлично |
| Решению соответствует большинству требований, позволяющих оценить его «отлично», но имеется один из приведенных ниже недостатков:<br>- решение в целом верное, но задача решена неоптимальным способом или имеются небольшие недочеты, которые обучаемый не смог устранить во время защиты работы.<br>- пояснительная записка имеет небольшие отклонения по объему, структуре или правильности оформления;<br>- во время защиты работы обучаемый не ответил на ряд вопросов относительно своего решения. | хорошо |
| Решению соответствует большинству требований, позволяющих оценить его «хорошо», но имеется один из приведенных ниже недостатков:<br>- имеются существенные ошибки в решении, например, получение неправильного ответа при определенном наборе входных данных;<br>- пояснительная записка имеет серьезные отклонения по объему, структуре или правильности оформления;<br>- во время защиты работы обучаемый не ответил на половину вопросов относительно своего решения. | удовлетворительно |
| Имеется как минимум один из нижеуказанных недостатков:<br>- решение не соответствует заданию;<br>- пояснительная записка отсутствует или содержит неструктурированный и неоформленный текст;<br>- во время защиты работы обучаемый не ответил на больше половины вопросов относительно своего решения. | неудовлетворительно |

### 12.2.2 Оценивание результатов тестирования обучающихся

Тестирование проводится по изученным разделам (темам) учебной дисциплины для выявления уровня знаний и умений обучающегося, мотивирования обучающегося к активизации самостоятельной работы по усвоению учебного материала, выявления и устранения пробелов в знаниях, повышения дисциплины и организации учебной деятельности обучающегося.

Тестирование проводится в рамках часов, отведенных на самостоятельную работу обучающихся во время, назначенное преподавателем. Для тестирования используется 10 вопросов из базы типовых тестовых заданий по изученным разделам (темам) учебной дисциплины.

При проведении контроля обучаемому предлагаются вопросы трех типов:

– выбор только одного ответа. Обучаемый выбирает один из предложенных вариантов ответа. В случае выбора неправильного ответа обучаемый получает за данный вопрос 0 баллов;

– выбор одного или нескольких ответов. Обучаемый вправе выбрать любое непустое множество ответов. За каждый правильный ответ обучаемый получает определенное количество баллов (суммарная оценка все правильных ответов вопроса – 1 балл). При выборе хотя бы одного неправильного ответа обучаемый получает за данный вопрос 0 баллов;

– ввод ответа с клавиатуры. Обучаемый вводит с клавиатуры ответ, правильный с его точки зрения. Введенный ответ подвергается операциям стандартизации (например, приводится текст к одному регистру) и сравнивается с эталонным(и) значением(ями). При отсутствии совпадения с каким-либо эталонным значением ответ считается неправильным и обучаемый получает за данный вопрос 0 баллов. При использовании вопросов данного типа особое внимание уделяется формулировке вопроса, в которой могут указываться правила формализации вводимого ответа.

Оценка формируется в соответствии с критериями таблицы:

| Максимальный процент ответа | Проходной процент ответа | Оценка |
|---|---|---|
| 100% | не менее 80% | отлично |
| 79,99% | не менее 60% | хорошо |
| 59,99% | не менее 35% | удовлетворительно |
| 34,99% | – | неудовлетворительно |

### 12.2.3 Оценивание в процессе наблюдения за работой на практическом занятии

Наблюдение за работой на практическом занятии – важнейший компонент контроля деятельности обучающихся и качества их обучения.

**Критерии оценки в процессе наблюдения за работой на практическом занятии:**

– самостоятельность выполнения задания и работы с методическими указаниями (учитывается работа в течение занятия, быстрота и способность нахождения в методических указаниях нужной информации);

– правильность выполнения работы (учитывается последовательность выполняемых действий, умение работать с инструментами и/или средствами вычислительной техники);

– рациональность использования времени, отведенного на задание (не одобряется затянутость выполнения работы с учетом индивидуальных особенностей обучающихся);

– активность обучающегося при выполнении практической работы (учитывается индивидуальная работа и работа в группе).

Критерии оценки по результатам наблюдения за работой на практическом занятии

| Критерий оценивания | Оценка |
|---|---|
| Достигнут правильный результат, работа выполнена самостоятельно и оптимальным способом | зачтено |
| Получен неправильный результат, или работа выполнена не самостоятельно, или работа выполнена неоптимальным способом | не зачтено |

### 12.2.4 Оценивание в процессе наблюдения за работой на лабораторном занятии

Наблюдение за работой на лабораторном занятии – важнейший компонент контроля деятельности обучающихся и качества их обучения.

**Критерии оценки в процессе наблюдения за работой на лабораторном занятии:**

– самостоятельность выполнения задания и работы с методическими указаниями (учитывается работа в течение занятия, быстрота и способность нахождения в методических указаниях нужной информации);

– правильность выполнения работы (учитывается последовательность выполняемых действий, умение работать с инструментами и/или средствами вычислительной техники);

– рациональность использования времени, отведенного на задание (не одобряется затянутость выполнения работы с учетом индивидуальных особенностей обучающихся);

– активность обучающегося при выполнении лабораторной работы (учитывается индивидуальная работа и работа в группе).

Критерии оценки по результатам наблюдения за работой на лабораторном занятии

| Критерий оценивания | Оценка |
|---|---|
| Достигнут правильный результат, работа выполнена самостоятельно и оптимальным способом | зачтено |
| Получен неправильный результат, или работа выполнена не самостоятельно, или работа выполнена неоптимальным способом | не зачтено |

**Задания и вопросы для текущего контроля знаний**

**обучающихся по разделам (темам) учебной дисциплины приведены в приложении.**

## 13 Промежуточная аттестация

**13.1 Оценочные материалы для промежуточной аттестации** обучающихся по учебной дисциплине предназначены для оценки степени достижения запланированных результатов обучения по завершению изучения учебной дисциплины в установленной учебным планом форме и позволяют определить качество усвоения изученного материала.

Итоговой формой контроля сформированности компетенций у обучающихся по учебной дисциплине является: курсовая работа; экзамен.

Оценочные материалы для промежуточной аттестации обучающихся включают: вопросы к защите курсовой работы; экзамен на основе результатов компьютерного тестирования.

### 13.2 Оценивание обучающегося на защите курсовой работы

| Критерии оценки выполнения курсовой работы | Оценка |
|---|---|
| Полное верное решение, задача решена рациональным способом, пояснительная записка имеет необходимый объем, структуру и правильность оформления, во время защиты работы обучаемый ответил на все вопросы относительно своего решения. | отлично |
| Решению соответствует большинству требований, позволяющих оценить его «отлично», но имеется один из приведенных ниже недостатков:<br>- решение в целом верное, но задача решена неоптимальным способом или имеются небольшие недочеты, которые обучаемый не смог устранить во время защиты работы.<br>- пояснительная записка имеет небольшие отклонения по объему, структуре или правильности оформления;<br>- во время защиты работы обучаемый не ответил на ряд вопросов относительно своего решения. | хорошо |
| Решению соответствует большинству требований, позволяющих оценить его «хорошо», но имеется один из приведенных ниже недостатков:<br>- имеются существенные ошибки в решении, например, получение неправильного ответа при определенном наборе входных данных;<br>- пояснительная записка имеет серьезные отклонения по объему, структуре или правильности оформления;<br>- во время защиты работы обучаемый не ответил на половину вопросов относительно своего решения. | удовлетворительно |
| Имеется как минимум один из нижеуказанных недостатков:<br>- решение не соответствует заданию;<br>- пояснительная записка отсутствует или содержит | неудовлетворительно |

| | |
|---|---|
| неструктурированный и неоформленный текст;<br>- во время защиты работы обучаемый не ответил на больше половины вопросов относительно своего решения. | |

Если допущены существенные недостатки в оформлении курсовой работы, опущен или не написан какой-либо раздел, имеются отступления от задания на курсовую работу, то такая курсовая работа возвращается обучающемуся на доработку.

### 13.3 Оценивание обучающегося на экзамене

| Максимальный процент ответа | Проходной процент ответа | Оценка |
|---|---|---|
| 100% | не менее 80% | отлично |
| 79,99% | не менее 60% | хорошо |
| 59,99% | не менее 35% | удовлетворительно |
| 34,99% | – | неудовлетворительно |

**Задания и вопросы для промежуточной аттестации обучающихся приведены в приложении.**

## 14 Учебно-методическое и информационное обеспечение учебной дисциплины

**а) литература:**

1 Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2024. — 213 с. — ISBN 978-5-534-16316-2. – URL: https://urait.ru/bcode/537332 (дата обращения: 28.04.2025);

2 Подбельский, В. В. Программирование. Базовый курс C# : учебник для вузов / В. В. Подбельский. — Москва : Издательство Юрайт, 2024. — 369 с. — ISBN 978-5-534-10616-9. – URL: https://urait.ru/bcode/536775 (дата обращения: 28.04.2025);

3 Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 151 с. — ISBN 978-5-534-16941-6. – URL: https://urait.ru/bcode/537385 (дата обращения: 28.04.2025).

**б) ресурсы информационно-телекоммуникационной сети «Интернет»:**

1 Электронный каталог : сайт / Научно-техническая библиотека СибГИУ. – Новокузнецк, [199 – ]. – URL: http://libr.sibsiu.ru. – URL: https://libr.sibsiu.ru;

2 Электронная библиотека // Научно-техническая библиотека СибГИУ : сайт. – Новокузнецк, [200 – ]. – URL: http://library.sibsiu.ru/LibrELibraryFullText.asp. – Режим доступа: для авторизир. пользователей. – URL: https://library.sibsiu.ru/LibrELibraryFullText.asp;

3 Консультант студента : электронно-библиотечная система / ООО «КОНСУЛЬТАНТ СТУДЕНТА». – Москва, [200 – ]. – URL: http://www.studentlibrary.ru. – Режим доступа: для авторизир. пользователей;

4 ЛАНЬ : электронно-библиотечная система : [коллекция «Единая профессиональная база знаний для технических вузов - Издательство Лань" ЭБС»] / ООО «Издательство ЛАНЬ». – Санкт-Петербург, [200 – ]. – URL: http://e.lanbook.com. – Режим доступа: для авторизир. пользователей;

5 Образовательная платформа ЮРАЙТ / ООО «Электронное издательство ЮРАЙТ». – Москва, [200 – ]. – URL: https://urait.ru. – Режим доступа: для авторизир. пользователей;

6 Университетская библиотека онлайн : электронно-библиотечная система / ООО «Директ-Медиа». – Москва, [200 – ]. – URL: https://biblioclub.ru. – Режим доступа: для авторизир. пользователей. – URL: http://www.biblioclub.ru;

7 НАУЧНАЯ ЭЛЕКТРОННАЯ БИБЛИОТЕКА eLIBRARY.RU : база данных / ООО «НЭБ». – Москва, [200 – ]. – URL: http://elibrary.ru. – Режим доступа: по подписке;

8 Электронные периодические издания ИВИС : универсальная база данных / ООО «ИВИС». – Москва, [200 – ]. – URL: http://eivis.ru. – Режим доступа: по подписке.

**в) лицензионное и свободно распространяемое программное обеспечение:**

– 7-Zip;
– Adobe Acrobat Reader;
– Astra Linux Special Edition;
– Kaspersky Endpoint Security;
– Microsoft Office;
– Microsoft Windows;
– OnlyOffice;
– Visual Studio;
– Р7-Офис.

**г) базы данных и информационно-справочные системы:**

1 ГАРАНТ : справочно-правовая система / ООО «Правовой центр «Гарант». – Кемерово, [200 – ]. – Режим доступа: компьютерная сеть Сиб. гос. индустр. ун-та.;

2 КонсультантПлюс : справочно-правовая система / ООО «Информационный центр АНВИК». – Новокузнецк, [199 – ]. – Режим доступа: компьютерная сеть библиотеки Сиб. гос. индустр. ун-та.;

3 Техэксперт : информационно-справочная система / ООО «Группа компаний «Кодекс». – Кемерово, [200 – ]. – Режим доступа: компьютерная сеть Сиб. гос. индустр. ун-та.

## 15 Материально-техническое обеспечение учебной дисциплины

Материально-техническое обеспечение учебной дисциплины включает учебные аудитории, оснащенные оборудованием, компьютерной техникой, и техническими средствами обучения, в том числе:
- учебную аудиторию для проведения занятий лекционного типа, оборудованную учебной доской, экраном и мультимедийным проектором;
- учебную аудиторию для проведения занятий семинарского типа (практических занятий), выполнения курсовых работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенную компьютерной техникой;
- помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду, научно-техническую библиотеку СибГИУ.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки (специальности) 09.03.01 «Информатика и вычислительная техника».

Составитель(и):

доцент Кожемяченко Вадим Иванович (кафедра прикладных информационных технологий и программирования).

Рабочая программа дисциплины рассмотрена и утверждена на заседании кафедры.

Согласована:

*Ген. директор ООО «Агит-плюс»*  *А.Ю. Рыжих*

должность, степень, звание представителя работодателя — инициалы, фамилия

## Приложение

## 1 Тестовое задание для входного контроля знаний обучающихся

Элементом списка является пара целых чисел. Отображение элемента осуществляется в формате «<число 1> / <число 2>». Требуется создать класс, обеспечивающий хранение элемента списка, организовать возможность добавления и удаления элементов списка, а также обеспечить нахождение суммы первых чисел элементов, у которых вторые числа отрицательные.

## 2 Задания и вопросы для текущего контроля знаний обучающихся по разделам (темам) учебной дисциплины

*Тема 1.2 Перегрузка операторов*

Вопрос 1.

Имеется описание класса:

```
class MyClass
{
  private int a, b;
  public static MyClass <...> +(MyClass op1, MyClass
op2)
  {
    ...
  }
}
```

Какое ключевое слово должно стоять вместо **<...>** для правильной перегрузки оператора «+»

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
class MyClass
{
  public int a, b;
  public  static  MyClass  operator  +(MyClass  op1,
MyClass op2)
  {
    MyClass cl = new MyClass();
    cl.a = op1.a + op2.a;
    cl.b = op1.b + op2.b;
    return op1;
  }
```

```
    public static MyClass operator +(MyClass op1, int
op2)
    {
      MyClass cl = new MyClass();
      cl.a = op1.a + op2;
      cl.b = op1.b + op2;
      return op1;
    }
  }

  MyClass cl1 = new MyClass() { a = 1, b = 2 };
  MyClass cl2 = new MyClass() { a = 3, b = 4 };
  <...>
```

Какие строки кода, подставленные вместо <...>, не содержат ошибок?

Метод ответа – выбор одного или нескольких ответов.

- `MyClass cl3 = cl1 + cl2;`
- `MyClass cl3 = cl1 + 5;`
- `MyClass cl3 = 5 + cl2;`
- `MyClass cl3 = 5 + 6;`

Вопрос 3.

Какие из операторов должны перегружаться попарно?

Метод ответа – выбор одного или нескольких ответов.

- `== и !=`
- `< и >`
- `+ и -`
- `* и /`

*Тема 1.3 Делегаты*

Вопрос 1.

Имеется фрагмент программы:

```
  delegate int MyDelegate(int a, int b);

  class MyClass
  {
    public int Method1(int c, int d) { return c + d; }
    public int Method2(int c, int d) { return c - d; }
  }

  MyClass cl = new MyClass();
  MyDelegate d = new MyDelegate(cl.Method1);
  int i = d(4, 5);
  d = cl.Method2;
  i += d(6, 7);
```

Какое значение будет в переменной i после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
delegate int MyDelegate(int a);

class MyClass
{
  public int Method1(int c) { return c + 2; }
  public int Method2(int c) { return c - 4; }
}

MyClass cl = new MyClass();
MyDelegate d = new MyDelegate(cl.Method1);
d += cl.Method2;
int i = 5;
i = d(i);
```

Каким будет значение переменной i после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 3.

Имеется фрагмент программы:

```
delegate int MyDelegate(int n);

MyDelegate d = delegate(int m)
{
  return m * 2;
};
```

Допустимо ли такое описание и определение объекта d?

Метод ответа – выбор только одного ответа.

- да, т.к. здесь используется анонимный метод
- нет, т.к. объекту типа «делегат» должен быть присвоен описанный метод, имеющий идентификатор
- да, т.к. здесь используется лямбда-выражение
- нет, т.к. идентификаторы параметров в описании делегата и в определении объекта d различны (n и m)

Вопрос 4.

Имеется фрагмент программы:

```
delegate MyClass1 MyDelegate(MyClass3 c);

class MyClass1
{
  public int a;
  public MyClass1 Method(MyClass1 c) { return new
MyClass1() { a = 5 }; }
}
class MyClass2 : MyClass1
{
  public MyClass2 Method(MyClass2 c) { return new
MyClass2() { a = 6 }; }
}
class MyClass3 : MyClass2
{
  public MyClass3 Method(MyClass3 c) { return new
MyClass3() { a = 7 }; }
}

MyClass1 c1 = new MyClass1();
MyClass2 c2 = new MyClass2();
MyClass3 c3 = new MyClass3();
<...>
```

В какой(их) из строк кода, подставленной(ых) вместо <...>, используется свойство контравариантности делегатов?

Метод ответа – выбор одного или нескольких ответов.

- MyDelegate d = c1.Method;
- MyDelegate d = c2.Method;
- MyDelegate d = c3.Method;
- свойство контравариантности не используется ни в одном из других вариантов ответов

*Тема 1.4 Анонимные методы и лямбда-выражения*

Вопрос 1.

Имеется фрагмент программы:

```
delegate int MyDelegate(int n);

MyDelegate d = m <...>
{
  int sum = 0;
  for (int i = 1; i <= m; i++)
    sum += i;
  return sum;
```

```
};
```

Какой код должен стоять вместо **<...>** при описании метода с использованием лямбда-выражения?

Метод ответа – ввод ответа.

*Тема 1.5 События*

Вопрос 1.

Имеется фрагмент программы:

```
delegate void MyEventHandler();

class RunEvent
{
  public event MyEventHandler MyEvent;
  public void OnMyEvent()
  {
    if (MyEvent != null)
      MyEvent();
  }
}

class MyClass
{
  public int i = 5;
  public void Run() { i += 2; }
}

RunEvent clRun = new RunEvent();
MyClass cl = new MyClass();
clRun.MyEvent += cl.Run;
```

Какая строка позволит убрать обработчик Run объекта cl из списка обработчиков события MyEvent объекта clRun?

Примечание: при вводе не использовать пробелы

Метод ответа – ввод ответа.

Вопрос 2.

Какова структура описания события, которому требуется выполнять дополнительные действия при добавлении или удалении обработчика событий?

Метод ответа – выбор одного или нескольких ответов.

```
- event <имя делегата> <имя события>
 {
   add { ... }
   remove { ... }
 }
```

- event <имя делегата> <имя события>
 {
   set { ... }
   get { ... }
 }
- event <имя делегата> <имя события>
 {
   add { ... }
   delete { ... }
 }
- событие, выполняющее дополнительные действия при добавлении или удалении обработчика событий, создать невозможно

*Тема 2.1 Общая схема построения универсальных типов*

Вопрос 1.

Каково основное назначение универсальных типов?

Метод ответа – выбор только одного ответа.

- однократная реализация некоторого алгоритма и ее последующее применение для объектов разных типов
- создание переменной, в которую можно записать любой тип данных
- создание переменной, в которую можно записать данные любого типа-значения
- создание базового класса, имеющего заданный набор методов, на основе которого будет создаваться классы-потомки с конкретной реализацией каждого из методов

Вопрос 2.

Имеется фрагмент программы:

```
class MyClass<T>
{
  T Value;
  public T Get() { return Value; }
  public void Set(T NewValue) { Value = NewValue; }
}
```

Что означает в заголовке класса блок <T>?

Метод ответа – выбор только одного ответа.

- при создании объекта данного класса вместо T требуется подставить реальный тип данных
- класс является наследником класса T
- класс реализует интерфейс T
- при создании объекта данного класса вместо T необходимо указать начальное значение поля Value

Вопрос 3.

Имеется заголовок универсального типа:

```
class MyClass<...>
```

Что должно быть указано вместо **<...>**, чтобы класс был универсальным с двумя параметрами типов t1 и t2?

Примечание: при воде не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 4.

Имеется описание класса:

```
class MyClass<T>
{
  T Value;
  public T Get() { return Value; }
  public void Set(T NewValue) { Value = NewValue; }
  public void Clear() { Value = <...> }
}
```

Что должно быть указано вместо **<...>**, чтобы полю было присвоено значение «по умолчанию» для его типа?

Примечание: при воде не использовать пробелы.

Метод ответа – ввод ответа.

*Тема 2.2 Ограничения, накладываемые в универсальных типах*

Вопрос 1.

Имеется заголовок класса:

```
class MyB<T> where <...>
```

Что должно быть указано вместо **<...>**, чтобы указать, что тип «T» должен быть наследником класса MyA или реализовывать интерфейс MyA?

Примечание: при воде не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 2.

Имеется заголовок класса:

```
class MyClass<T> where <...>
```

Что должно быть указано вместо **<...>**, чтобы указать, что тип «T» должен быть ссылочным типом?

Метод ответа – ввод ответа.

*Тема 2.3 Параметры типы в методах*

Вопрос 1.

Имеется фрагмент программы:

```
class MyClass1
{
  public int n;
}

class MyClass2
{
  public int m;
}

class MyClass3<T> where T : MyClass1
{
  public T k;
}
```

Какая(ие) строка(и) по созданию объекта допустима(ы)?

Метод ответа – выбор одного или нескольких ответов.

- `MyClass3<MyClass1> cl = new MyClass3<MyClass1>();`
- `MyClass3<MyClass2> cl = new MyClass3<MyClass2>();`
- `MyClass3<int> cl = new MyClass3<int>();`
- `MyClass3<string> cl = new MyClass3<string>();`

*Тема 3.1 Введение в LINQ*

Вопрос 1.

Какое(ие) из описаний и определений переменной анонимного типа выполнено(ы) правильно?

Метод ответа – выбор одного или нескольких ответов.

- `class Class1 {...}`
  `var a = new { b = new Class1(), c = 5 };`
- `var a = new { string s = "abc"};`
- `var a = new var() { b = 1, c = 2.3 };`
- `var a = { b = 1, c = 2.3 };`
- `double a = new { b = 1.2};`

Вопрос 2.

Какое(ие) из нижеприведенных высказываний можно отнести к методам расширения?

Метод ответа – выбор одного или нескольких ответов.

- должен быть статическим
- должен иметь как минимум один параметр с модификатором this
- не может иметь параметров

– должен иметь в качестве возвращаемого типа void
– должен иметь в качестве возвращаемого типа тот тип, который он расширяет

Вопрос 3.

Имеется фрагмент программы:

```
class Class1
{
  public int D() { return 1; }
}

static class Class2
{
  public static int D(this Class1 p, int n) { return n + 2; }
}

Class1 c = new Class1();
int i = c.D(3);
```

Каким будет значение переменной i?

Метод ответа – ввод ответа.

*Тема 3.2 Построение запросов на LINQ*

Вопрос 1.

Имеется фрагмент программы:

```
int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
int[] b = (from c in a
           where c / 2 >= 3
           select c * 2).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Примечание: при вводе перечислять числа через запятую, не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
var b = from c in a
        where c > 3
        select c;
a[3] = 9;
string s = "";
```

```
foreach (var el in b)
  s += el + ", ";
```

Какое значение будет содержать переменная s после выполнения фрагмента?

Метод ответа – выбор только одного ответа.

- “5, 7, 9, 4, 8, “
- “5, 7, 4, 8, “
- “2, 4, 1, 8, “
- “4, 5, 7, 8, 9, “

Вопрос 3.

Имеется фрагмент программы:

```
int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
int[] b = (from int c in a
           where c > 2 && c % 2 == 0
           select c).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Примечание: при вводе перечислять числа через запятую, не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 4.

Имеется фрагмент программы:

```
class People
{
  public string Name;
  public int Age;
}

People[] a = {
             new People() { Name="Иванов", Age=18 },
             new People() { Name="Петров", Age=30 },
             new People() { Name="Антонов", Age=18 },
             new People() { Name="Борисов", Age=28 },
             new People() { Name="Яковлев", Age=21 }
             };

var b = from c in a
        ... c.Age == 18
        select c;
```

Что должно быть вписано вместо ..., чтобы запрос отобрал людей, значение поля Age которых равно 18?

Метод ответа – ввод ответа.

Вопрос 5.

Имеется фрагмент программы:

```
class People
{
  public string Name;
  public int Age;
}

People[] a = {
                new People() { Name="Иванов", Age=18
},
                new People() { Name="Петров", Age=30
},
                new People() { Name="Антонов", Age=18
},
                new People() { Name="Борисов", Age=28
},
                new People() { Name="Иванов", Age=21 }
              };

var b = (from c in a
         group c by c.Age).ToArray();

textBox1.Clear();
foreach (var g in b)
{
  textBox1.AppendText(g.Key + "\n");
  foreach (var p in g)
    textBox1.AppendText("  " + p.Name + "\n");
}
```

Что будет выведено в компонент textBox1?

Метод ответа – выбор только одного ответа.

```
- 18
    Иванов
    Антонов
  30
    Петров
  28
    Борисов
```

```
  21
    Иванов
- 18
    Иванов
    Антонов
  21
    Иванов
  28
    Борисов
  30
    Петров
- Иванов  18
  Антонов  18
  Петров  30
  Борисов  28
  Иванов  21
```

- фрагмент не будет выполнен, т.к. в классе People нет поля Key

Вопрос 6.

Имеется фрагмент программы

```
class City
{
  public int ID;
  public string Name;
}

class People
{
  public string Name;
  public int Age;
  public int IDCity;
}

City[] Cities = {
            new City() {ID=1, Name="Новокузнецк"},
            new City() {ID=2, Name="Новосибирск"},
            new City() {ID=3, Name="Красноярск"},
            new City() {ID=4, Name="Омск"}
                };

People[] Peoples = {
  new People() { Name="Иванов", Age=18, IDCity=1 },
  new People() { Name="Петров", Age=30, IDCity=4 },
  new People() { Name="Антонов", Age=18, IDCity=2 },
```

```
      new People() { Name="Борисов", Age=28, IDCity=2 },
      new People() { Name="Иванов", Age=21, IDCity=1 }
                     };

    var r = (from c in Cities
             from p in Peoples
             where p.Age == 18
             select new { nC = c.Name, nP = p.Name
}).ToArray();
```

Сколько элементов будет в массиве r?

Метод ответа – ввод ответа.

Вопрос 7.

Имеется фрагмент программы:

```
    class People
    {
      public string Name;
      public int Age;
    }

    People[] a = {
             new People() { Name="Иванов", Age=18 },
             new People() { Name="Петров", Age=30 },
             new People() { Name="Антонов", Age=18 },
             new People() { Name="Борисов", Age=28 },
             new People() { Name="Иванов", Age=21 }
                 };

    string[] b = (from c in a
                  where c.Name.Length == 6
                  orderby c.Age
                  select c.Name).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Метод ответа – выбор только одного ответа.

- { {"Иванов", 18}, {"Иванов", 21}, {"Петров", 30} }
- { "Иванов", "Иванов", "Петров" }
- { {"Иванов", 21}, {"Иванов", 18}, {"Петров", 30} }
- { {"Иванов", 18}, {"Петров", 30}, {"Иванов", 21} }
- { "Антонов", "Борисов", "Иванов", "Иванов", "Петров" }

*Тема 3.3 Использование методов интерфейса перечислителя*

Вопрос 1.

Имеется фрагмент программы:

```
double[] a = { 1, 2, 3, 4, 5 };
<...> b = a.Select(value => -value);
```

Какой тип данных должен быть указан вместо <...>?

Примечание: при вводе не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
int[] a = { 5, 2, 7, 6, 9 };
int b = a.Count(c => c * 3 > 17);
```

Какой значение будет иметь переменная b после выполнения фрагмента?

Метод ответа – ввод ответа.

*Раздел 4 Регулярные выражения*

Вопрос 1.

Имеется фрагмент программы:

```
string s = "abacbacde";
string pattern = @"[ab]"1;
MatchCollection res = Regex.Matches(s, pattern);
```

Сколько элементов будет в коллекции «res»?

Метод ответа – ввод ответа.

## 3 Вопросы к экзамену для промежуточной аттестации

*Тема 1.2 Перегрузка операторов*

Вопрос 1.

Имеется описание класса:

```
class MyClass
{
  private int a, b;
  public static int operator +(MyClass op1, int op2)
  {
    ...
  }
  public static int operator +(<1> op1, <2> op2)
  {
    ...;
```

```
  }
}
```

Какие типы, подставленные вместо **<1>** и **<2>** сделают возможным использование следующего кода (**если решения нет, то в ответе указать слово «нет» без кавычек**)?

```
MyClass cl = new MyClass();
int n = 5 + cl;
```

Примечание: при вводе перечислить типы через запятую и не использовать пробелы.

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
class MyClass
{
  public int a, b;
  public static MyClass operator +(MyClass op1, int
op2)
  {
    MyClass cl = new MyClass();
    cl.a = op1.a + op2;
    cl.b = op1.b + op2;
    return cl;
  }
  public static int operator +(int op1, MyClass op2)
  {
    return op1 + op2.a + op2.b;
  }
}

MyClass cl1 = new MyClass() { a = 1, b = 2 };
MyClass cl2 = new MyClass() { a = 3, b = 4 };
int n = 5;
<...>
```

Какие строки кода, подставленные вместо <...>, не содержат ошибок?

Метод ответа – выбор одного или нескольких ответов.

- `MyClass cl = cl1 + n;`
- `int m = n + cl1;`

- int m = cl1 + n;
- MyClass cl = n + cl1;
- MyClass cl = cl1 + cl2;

Вопрос 3.

Какие из операторов должны перегружаться попарно?

Метод ответа – выбор одного или нескольких ответов.

- <= и >=
- < и >
- + и -
- * и /

*Тема 1.3 Делегаты*

Вопрос 1.

Имеется фрагмент программы:

```
delegate int MyDelegate(int a, int b);

class MyClass
{
  public static int Method1(int a, int b) { return a
+ b; }
  public int Method2(int c, int d) { return c * d; }
  public int Method3(int e) { return e * 2; }
}

MyClass cl = new MyClass();
<...>
```

Какая(ие) из строк, подставленная(ые) вместо <...>, допустима(ы)?

Метод ответа – выбор одного или нескольких ответов.

- MyDelegate d = new MyDelegate(MyClass.Method1);
- MyDelegate d = new MyDelegate(cl.Method2);
- MyDelegate d = new MyDelegate(cl.Method3);
- MyDelegate d = new MyDelegate(cl.Method1);
- MyDelegate d = new MyDelegate(MyClass.Method2);

Вопрос 2.

Имеется фрагмент программы:

```
delegate int MyDelegate(int a, int b);

class MyClass
{
  public int Method1(int a, int b) { return a + b; }
  public int Method2(int c, int d) { return c - d; }
}
```

```
MyClass cl = new MyClass();
MyDelegate d = new MyDelegate(cl.Method1);
int i = d(4, 5);
d = cl.Method2;
i = d(i, 6);
```

Какое значение будет в переменной i после выполнения фрагмента?

Метод ответа – вод ответа.

Вопрос 3.

Имеется фрагмент программы:

```
delegate void MyDelegate(ref int a);

class MyClass
{
  public void Method1(ref int c) { c += 2; }
  public void Method2(ref int c) { c -= 4; }
}

MyClass cl = new MyClass();
MyDelegate d = new MyDelegate(cl.Method1);
d += cl.Method2;
d -= cl.Method2;
d -= cl.Method1;
int i = 5;
d(ref i);
```

Каким будет значение переменной i после выполнения фрагмента?

Примечание: если фрагмент не будет выполнен, то ввести слово «нет» без кавычек.

Метод ответа – ввод ответа.

Вопрос 4.

Имеется фрагмент программы:

```
delegate MyClass1 MyDelegate(MyClass3 c);

class MyClass1
{
  public int a;
  public MyClass1 Method(MyClass1 c) { return new MyClass1() { a = c.a }; }
}
```

```
    class MyClass2 : MyClass1
    {
      public MyClass1 Method(MyClass2 c) { return new
MyClass2() { a = c.a * 2 }; }
    }

    class MyClass3 : MyClass2
    {
      public MyClass1 Method(MyClass3 c) { return new
MyClass3() { a = c.a * 3 }; }
    }

    MyClass1 c1 = new MyClass1();
    MyClass2 c2 = new MyClass2();
    MyClass3 c3 = new MyClass3();
    <...>
```

В какой(их) из строк кода, подставленной(ых) вместо <...>, используется свойство ковариантности делегатов?

Метод ответа – выбор одного или нескольких ответов.

- свойство ковариантности не используется ни в одном из других вариантов ответов
- MyDelegate d = c1.Method;
- MyDelegate d = c2.Method;
- MyDelegate d = c3.Method;

*Тема 1.4 Анонимные методы и лямбда-выражения*

Вопрос 1.

Имеется фрагмент программы:

```
delegate int MyDelegate(int start, int end);


MyDelegate d = <...>
 {
   ...
 };
```

Что надо ввести вместо **<...>**, чтобы создать делегата с использованием лямбда-выражения.

Примечание: имена переменных и порядок их расположения в лябмбда-выражении такой же, как и в делегате. Пробелы не использовать.

Метод ответа – ввод ответа.

*Тема 1.5 События*

Вопрос 1.

Имеется фрагмент программы:

```
delegate void MyEventHandler();

class RunEvent
{
  public event MyEventHandler MyEvent;
  public void OnMyEvent()
  {
    if (MyEvent != null)
      MyEvent();
  }
}

class MyClass
{
  public int i = 5;
  public void Run() { i += 2; }
}

RunEvent clRun = new RunEvent();
MyClass cl = new MyClass();
clRun.MyEvent += cl.Run;
clRun.MyEvent += cl.Run;
clRun.OnMyEvent();
```

Сколько обработчиков события будет вызвано при выполнении строки clRun.OnMyEvent(); (т.е. сколько вызовов методов выполнится)?

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
delegate void MyEventHandler();

class RunEvent
{
  private MyEventHandler ListHandler;
  public event MyEventHandler MyEvent
```

```
  {
    add { ListHandler += value; }
    remove { ListHandler -= value; }
  }
  public void OnMyEvent()
  {
    <...>
  }
}
```

Какой(ие) из приведенных вариантов ответа, подставленный(ые) вместо <...>, допустим(ы) (т.е. не вызовут ошибку при компиляции)?

Метод ответа – выбор одного или нескольких ответов.

- `if (ListHandler != null) ListHandler();`
- `if (MyEvent != null) MyEvent();`
- `if (ListHandler != null) MyEvent();`
- `if (MyEvent != null) ListHandler();`

*Тема 2.1 Общая схема построения универсальных типов*

Вопрос 1.

```
class MyClass<T>
{
  T Value;
  public T Get() { return Value; }
  public void Set(T NewValue) { Value = NewValue; }
}

MyClass<double> cl = new MyClass<double>();
<...> p = cl.Get();
```

Какой тип данных должен быть подставлен вместо <...>?

Метод ответа – ввод ответа.

Вопрос 2.

Имеется заголовок универсального типа:

```
class MyClass<T1, T2>
```

Имеются ли ошибки в заголовке?

Метод ответа – выбор только одного ответа.

- нет
- да, т.к. у параметров T1 и T2 не указаны типы

– да, т.к. между T1 и T2 должна быть не запятая, а точка с запятой

– да, т.к. между T1 и T2 должна быть не запятая, а двоеточие

Вопрос 3.

Имеется описание класса:

```
class MyClass<T>
{
  private T Value;
  public T Get() { return Value; }
  public void Set(T NewValue) { Value = NewValue; }
  public void Clear() { Value = default(T); }
}

MyClass<int> cl = new MyClass<int>();
int n = 1;
cl.Clear();
n = cl.Get();
```

Какое значение будет у переменной n после выполнения фрагмента?

Метод ответа – ввод ответа.

*Тема 2.2 Ограничения, накладываемые в универсальных типах*

Вопрос 1.

Имеется заголовок класса:

```
class MyB<T> where <...>
```

Что нужно записать в месте <...>, чтобы указать, что параметр типа T должен иметь конструктор без параметров?

**Примечание: ввод производить без пробелов.**

Метод ответа – ввод ответа.

Вопрос 2.

Имеется фрагмент программы:

```
class MyClass1
{
  public int n;
}

class MyClass2
{
  public int m;
```

```
}

class MyClass3<T> where T : MyClass1
{
  public T k;
}
```

Какая(ие) строка(и) по созданию объекта допустима(ы)?

Метод ответа – выбор одного или нескольких ответов.

- `MyClass3<MyClass1> cl = new MyClass3<MyClass1>();`
- `MyClass3<MyClass2> cl = new MyClass3<MyClass2>();`
- `MyClass3<int> cl = new MyClass3<int>();`
- `MyClass3<string> cl = new MyClass3<string>();`

*Тема 2.3 Параметры типы в методах*

Вопрос 1.

Имеется фрагмент программы:

```
class MyClass
{
  public void Method<T>(T arg, out T res) where T :
struct
  {
    ...
  }
}

MyClass cl = new MyClass();
int i;
double d;
string s;
```

Какой(ие) вызов(ы) метода cl.Method допустим(ы)?

Метод ответа – выбор одного или нескольких ответов.

- `cl.Method(5, out i);`
- `cl.Method<double>(5, out i);`
- `cl.Method("abc", out s);`
- `cl.Method<int>(5, out d);`

Вопрос 2.

Имеется фрагмент программы:

```
class MyClass
{
  public void Method<T>(T arg, out T res) where T :
<...>
  {
```

```
        ...
      }
    }

    MyClass cl = new MyClass();
    int i;
    double d;
```

Что должно быть введено вместо **<...>**, чтобы в качестве параметра “T” можно было указать только ссылочный тип данных?

Метод ответа – ввод ответа.

*Тема 3.1 Введение в LINQ*

Вопрос 1.

Какое(ие) из описаний и определений переменной анонимного типа выполнено(ы) правильно?

Метод ответа – выбор одного или нескольких ответов.

- `var a = new { b = 1, c = 2.3 };`
- `var a = new { int b = 1, double c = 2.3 };`
- `var a = new var() { b = 1, c = 2.3 };`
- `var a = { b = 1, c = 2.3 };`
- `int a = new { b = 1};`

Вопрос 2.

Какое(ие) из нижеприведенных высказываний можно отнести к методам расширения?

Метод ответа – выбор одного или нескольких ответов.

- должен быть статическим
- вызывается через объект, а не через класс
- должен быть виртуальным
- должен быть абстрактным
- вызывается через класс, а не через объект

Вопрос 3.

Имеется метод расширения типа int со следующей сигнатурой

```
public static string Method(this int p)
```

Имеется фрагмент программы

```
int i = 5;
  string s = <1>Method(<2>);
```

Что должно быть подставлено вместо **<1>** и **<2>**, чтобы использовать переменную i при вызове метода?

Примечание: значения разделять запятой, пробелы не использовать. Если вместо какого-то блока ничего ставить не надо, то указать слово «нет» (без кавычек).

Метод ответа – ввод ответа.

*Тема 3.2 Построение запросов на LINQ*

Вопрос 1.

Имеется фрагмент программы:

```
class People
{
  public string Name;
  public int Age;
}

People[] a = {
          new People() { Name="Иванов", Age=18 },
          new People() { Name="Петров", Age=30 },
          new People() { Name="Антонов", Age=18 },
          new People() { Name="Борисов", Age=28 },
          new People() { Name="Иванов", Age=21 }
             };

var b = (from c in a
         group c by c.Name).ToArray();

textBox1.Clear();
foreach (var g in b)
{
  textBox1.AppendText(g.Key + "\n");
  foreach (var p in g)
    textBox1.AppendText("  " + p.Age + "\n");
}
```

Что будет выведено в компонент textBox1?

Метод ответа – выбор только одного ответа.

```
- Иванов
    18
    21
  Петров
    30
  Антонов
    18
  Борисов
    28
```

- Антонов
  18
  Борисов
  28
  Иванов
  18
  21
  Петров
  30
- Иванов  18
  Иванов  21
  Петров  30
  Антонов  18
  Борисов  28
- фрагмент не будет выполнен, т.к. в классе People нет поля Key

Вопрос 2.

Имеется фрагмент программы

```
class City
{
  public int ID;
  public string Name;
}

class People
{
  public string Name;
  public int Age;
  public int IDCity;
}

City[] Cities = {
        new City() {ID=1, Name="Новокузнецк"},
        new City() {ID=2, Name="Новосибирск"},
        new City() {ID=3, Name="Красноярск"},
        new City() {ID=4, Name="Омск"}
                };

People[] Peoples = {
  new People() { Name="Иванов", Age=18, IDCity=1 },
  new People() { Name="Петров", Age=30, IDCity=4 },
  new People() { Name="Антонов", Age=18, IDCity=2 },
  new People() { Name="Борисов", Age=28, IDCity=2 },
  new People() { Name="Иванов", Age=21, IDCity=1 }
```

```
                    };

    var r = (from c in Cities
             from p in Peoples
             select new { nC = c.Name, nP = p.Name
}).ToArray();
```

Сколько элементов будет в массиве r?

Метод ответа – ввод ответа.

Вопрос 3.

Имеется фрагмент программы:

```
    class People
    {
      public string Name;
      public int Age;
    }

    People[] a = {
              new People() { Name="Иванов", Age=18 },
              new People() { Name="Петров", Age=30 },
              new People() { Name="Антонов", Age=18 },
              new People() { Name="Борисов", Age=28 },
              new People() { Name="Иванов", Age=21 }
                 };

    People[] b = (from c in a
                  where c.Name.Length == 6
                  orderby c.Name
                  select c).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Метод ответа – выбор только одного ответа.

- { {"Иванов", 18}, {"Иванов", 21}, {"Петров", 30} }
- { "Иванов", "Иванов", "Петров" }
- { {"Иванов", 21}, {"Иванов", 18}, {"Петров", 30} }
- { {"Иванов", 18}, {"Петров", 30}, {"Иванов", 21} }
- { "Антонов", "Борисов", "Иванов", "Иванов", "Петров" }

Вопрос 4.

Имеется фрагмент программы:

```
    int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
    int[] b = (from c in a
```

```
            where c / 2 >= 2
            select c % 2).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Примечание: значения элементов массива разделять запятой, пробелы не ставить.

Метод ответа – ввод ответа.

Вопрос 5.

Имеется фрагмент программы:

```
int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
var b = from c in a
        where c > 2
        select c;
a[3] = 9;
string s = "";
foreach (var el in b)
  s += el + ", ";
```

Какое значение будет содержать переменная s после выполнения фрагмента?

Метод ответа – выбор только одного ответа.

- “5, 3, 7, 9, 4, 8, “
- “5, 3, 7, 4, 8, “
- “1, 2, 4, 1, 8, “
- “3, 4, 5, 7, 8, 9, “

Вопрос 6.

Имеется фрагмент программы:

```
int[] a = { 5, 3, 7, 1, 2, 4, 1, 8 };
int[] b = (from int c in a
           where c > 5 || c < 2
           select c).ToArray();
```

Какие значения будет содержать массив b после выполнения фрагмента?

Примечание: значения элементов массива разделять запятой, пробелы не ставить.

Метод ответа – ввод ответа.

Вопрос 7.

Имеется фрагмент программы:

```
class People
{
```

```
  public string Name;
  public int Age;
}

People[] a = {
         new People() { Name="Иванов", Age=18 },
         new People() { Name="Петров", Age=30 },
         new People() { Name="Антонов", Age=18 },
         new People() { Name="Борисов", Age=28 },
         new People() { Name="Яковлев", Age=21 }
             };

var b = from c in a
        where ...>'К'
        select c;
```

Что должно быть вписано вместо ..., чтобы запрос отобрал людей, значение поля Name которых начинается на букву, расположенную в русском алфавите после буквы «К»?

Примечание: при вводе не использовать пробелы.

Метод ответа – ввод ответа.

*Тема 3.3 Использование методов интерфейса перечислителя*

Вопрос 1.

Имеется фрагмент программы:

```
int[] a = { 1, 2, 3, 4, 5 };
<...> b = a.Select(value => -value);
```

Какой тип данных должен быть указан вместо <...>?

Метод ответа – выбор только одного ответа.

- IEnumerable<int>
- int[]
- int
- IEnumerable

Вопрос 2.

Имеется фрагмент программы:

```
int[] a = { 5, 2, 7, 6, 9 };
int b = a.Count();
```

Какой значение будет иметь переменная b после выполнения фрагмента?

Метод ответа – ввод ответа.

*Раздел 4 Регулярные выражения*

Вопрос 1.

Имеется фрагмент программы:

```
string s = "a.1b2.3c45.6d";
string pattern = @"\d?\.\d";
MatchCollection res = Regex.Matches(s, pattern);
```

Каким будет значение первого элемента коллекции?

Примечание: при необходимости используйте кавычки.

Метод ответа – ввод ответа.

## 4 Выполнение практических работ

Задания для практических работ, теоретический и методический материал для самостоятельного изучения и вопросы для собеседования представлены в методических указаниях к практическим занятиям.

## 5 Выполнение лабораторных работ

Задания для лабораторных работ, теоретический и методический материал для самостоятельного изучения и вопросы для собеседования представлены в методических указаниях к лабораторным работам.

## 6 Выполнение курсовой работы

В рамках учебной дисциплины предусмотрена самостоятельная работа обучающихся, результатом выполнения которой является оформленная в соответствии с требованиями пояснительная записка курсовой работы по заданной теме.

Примерный перечень тем курсовых работ представлен в рабочей программе дисциплины. Задание и методические указания к выполнению курсовой работы размещены в СУО «Moodle».

Перед выдачей темы курсовой работы обучающийся знакомится с графиком его выполнения и критериями оценки. Выполненная пояснительная записка курсовой работы загружается обучающимся в СУО «Moodle».

## 7 Вопросы к защите курсовой работы

Вопросы к защите курсовой работы формируются индивидуально для каждого обучаемого по результатам изучения представленной работы.

Примерный перечень вопросов:

1. Поясните назначение класса ___ (например, класса «Автомобиль»).

2. Поясните назначение членов класса ___ (например, класса «Автомобиль»).

3. Поясните схему функционирования моделируемой задачи.

4. Поясните, кто генерирует и кто является получателем события ___ (например, события «Новый автомобиль»).

5. Поясните назначение данного фрагмента кода.

6. Поясните логику данного фрагмента кода.

7. Поясните, как объекты класса ___ узнают о ___ (например, как объекты класса «Очередь автомобилей» узнают о том, что топливная колонка освободилась).

8. Покажите, где у вас в коде выполнена реализация события ___ (например, события «Новый автомобиль»).

9. Покажите, где у вас в коде выполнена реакция объектов на событие ___ (например, на событие «Новый автомобиль»).

10. Поясните, каким образом взаимодействуют объекты классов ___ и ___ (например, классов «Очередь автомобилей» и «Топливная колонка»).