

УТВЕРЖДАЮ

Зам. директора по УВР
Ур СЭИ (филиал) ОУП ВО «АТиСО»
_____ О.В. Зубкова
« 10» июня 2020 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
(МОДУЛЯ)**

Основы программирования
(название дисциплины в соответствии с учебным планом)

**СПЕЦИАЛЬНОСТЬ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО
ОБРАЗОВАНИЯ**

09.02.05 Прикладная информатика (по отраслям)
(код профессии, специальности СПО)

Техник-программист
(наименование квалификации)

Кафедра: Гуманитарных, естественнонаучных и математических дисциплин

Разработчики программы: Мадудин В.Н., к.т.н., доцент

Челябинск -2020

Оглавление

| | |
|---|----|
| 1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ | 3 |
| 1.1. Область применения рабочей программы учебного предмета, курса, дисциплины (модуля) | 3 |
| 1.2. Цели и задачи учебной дисциплины | 3 |
| 1.3. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена | 3 |
| 1.4. Требования к результатам освоения учебной дисциплины | 3 |
| 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ | 3 |
| 2.1. Объем учебной дисциплины и виды учебной работы | 4 |
| 2.2. Тематический план и содержание учебной дисциплины | 4 |
| 3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ | 7 |
| 3.1 Требования к минимальному материально-техническому обеспечению | 7 |
| 3.2 Информационное обеспечение реализации программы | 7 |
| 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ | 8 |
| 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ | 10 |
| 6. ОЦЕНОЧНЫЕ СРЕДСТВА И КОНТРОЛЬНО-ИЗМЕРИТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ | 11 |
| 6.1 Оценочные средства для проведения текущего контроля по учебной дисциплине | 11 |
| 6.2 Контрольно-измерительные материалы для проведения текущего контроля по учебной дисциплине | 13 |
| 7. ОЦЕНОЧНЫЕ СРЕДСТВА И КОНТРОЛЬНО-ИЗМЕРИТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ | 14 |
| 7.1. Оценочные средства для проведения промежуточной аттестации | 14 |
| по учебной дисциплине | 14 |
| 7.2. Контрольно-измерительные материалы для проведения промежуточной аттестации по учебной дисциплине | 20 |
| 8. ПРОВЕРКА СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ | 21 |

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1. Область применения рабочей программы учебного предмета, курса, дисциплины (модуля)

Рабочая программа учебной дисциплины «Основы программирования» является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности СПО 09.02.05 «Прикладная информатика (по отраслям)», квалификация Техник-программист.

1.2. Цели и задачи учебной дисциплины

– Цель изучения учебной дисциплины: сформировать понятийный аппарат, знания об основных этапах решения задач на ПК; о современных интегрированных средах для решения экономических задач.

Задачи изучения учебной дисциплины:

- изучить современные интегрированные среды для решения экономических задач;
- привить практические навыки разработки алгоритмов, кодирования, тестирования и отладки программ с использованием языка программирования высокого уровня C# / Python.

1.3. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена

Учебная дисциплина «Основы программирования» в программе подготовки специалистов среднего звена относится к дисциплинам профессионального цикла (МДК.02.02) специальности 09.02.05 «Прикладная информатика (по отраслям)».

Учебная дисциплина «Основы программирования» базируется на знаниях, умениях и навыках, полученных студентами при изучении дисциплины ОУДП.01 «Информатика».

Знания, умения и навыки, полученные студентами при изучении данной дисциплины, будут использованы при изучении дисциплин профессиональных модулей ПМ 03 – ПМ 04, а также при прохождении производственной и преддипломной практики.

1.4. Требования к результатам освоения учебной дисциплины

В результате освоения дисциплины обучающийся должен освоить следующие компетенции:

ПК 2.2. Разрабатывать и публиковать программное обеспечение и информационные ресурсы отраслевой направленности со статическим и динамическим контентом на основе готовых спецификаций и стандартов.

ПК 2.3. Проводить отладку и тестирование программного обеспечения отраслевой направленности.

В результате изучения дисциплины студент должен:

иметь практический опыт в:

- использовании IDE MS VisualStudio, языка программирования высокого уровня C# или Python на уровне разработчика приложений.

знать:

- 31 основы структурного и объектно-ориентированного подхода к разработке программ,
- 32 об инкапсуляции, наследовании и полиморфизме, а также событийно-управляемых приложениях,
- 33 назначение, основные свойства, методы и события стандартных элементов управления.

уметь:

- У1 использовать интегрированные среды разработки, языки программирования высокого уровня C# / Python и библиотеки классов для создания, тестирования и отладки консольных и оконных приложений.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

| Вид учебной работы | Объем часов 2г10м | Объем часов 3г10м |
|---|--|--|
| Максимальная учебная нагрузка (всего) | 286 | 286 |
| Обязательная аудиторная учебная нагрузка (всего) | 111 | 111 |
| В том числе: | | |
| теоретическое обучение | 56 | 56 |
| практические занятия | 55 | 55 |
| Самостоятельная работа обучающегося (всего) | 175 | 175 |
| В том числе: | | |
| работа по темам | 175 | 175 |
| подготовка докладов по темам | - | - |
| Итоговые аттестации | ДФК 1 семестр Зачет 2 семестр | ДФК 3 семестр Зачет 4 семестр |

2.2. Тематический план и содержание учебной дисциплины

| Наименование разделов и тем | Содержание учебного материала | Объем часов 2г10м/3г10м | Теоретические занятия | Практические занятия | Самостоятельная работа студента | Уровень освоения | Коды формируемых компетенций |
|------------------------------|--|----------------------------|-----------------------|----------------------|---------------------------------|------------------|------------------------------|
| | | | | | | | |
| Тема 1. Алгоритмы | Содержание учебного материала | 40 | 11 | 11 | 18 | 2 | ПК 2.2, ПК 2.3 |
| | Лекционные занятия Свойства и виды алгоритмов. Исполнители алгоритмов (назначение, среда, режим работы, система команд). Компьютер как формальный исполнитель алгоритмов. Блок-схемы, графические элементы. Базовые алгоритмические конструкции: СЛЕДОВАНИЕ, РАЗВИЛКА, ПОВТОРЕНИЕ. Использование псевдокода. Разработка алгоритмов путем пошаговой детализации. Практические занятия Решение задач на составление блок-схем. Самостоятельная работа изучение теоретического материала. | | | | | | |
| Тема 2. | Содержание учебного материала | 40 | 11 | 11 | 18 | 2 | ПК |

| | | | | | | | |
|--|--|----|----|----|----|---|-----------------------------------|
| <p>Системы счисления. Кодирование данных.</p> | <p>Лекционные занятия Единицы измерения количества информации. Непозиционные и позиционные системы счисления. 2-, 8- и 16-ричные системы счисления, перевод чисел из одной системы в другую. Кодирование символов, таблицы символов ASCII, ANSI, Unicode. Кодирование графической информации. Глубина цвета и количество отображаемых цветов. Цветовые модели RGB и CMYK.</p> <p>Практические занятия Решение задач на перевод данных из одной системы счисления в другую.</p> <p>Самостоятельная работа изучение теоретического материала.</p> | | | | | | <p>2.2, ПК 2.3</p> |
| <p>Тема 3. Структура программы на C# / Python. Ввод/вывод. Структуры управления</p> | <p>Содержание учебного материала</p> <p>Лекционные занятия Структура программы на C# / Python. Простейший ввод/вывод. Структуры управления: ветвящиеся и циклические алгоритмы и программы.</p> <p>Практические занятия Решение задач на реализацию ветвящихся и циклических алгоритмов.</p> <p>Самостоятельная работа изучение теоретического материала.</p> | 40 | 11 | 11 | 18 | 2 | <p>ПК 2.2, ПК 2.3</p> |
| <p>Тема 4. Массивы</p> | <p>Содержание учебного материала</p> <p>Лекционные занятия Одномерные, многомерные и ступенчатые массивы</p> <p>Практические занятия (лабораторные работы) Решение задач по работе массивами.</p> <p>Самостоятельная работа изучение теоретического материала.</p> | 40 | 11 | 11 | 18 | 2 | <p>ПК 2.2, ПК 2.3</p> |
| <p>Тема 5. Рекурсия</p> | <p>Содержание учебного материала</p> <p>Лекционные занятия Рекурсивные алгоритмы и программы</p> <p>Практические занятия Решение задач на реализацию рекурсивных алгоритмов.</p> <p>Самостоятельная работа изучение теоретического материала.</p> | 40 | 11 | 11 | 18 | 2 | <p>ПК 2.2, ПК 2.3</p> |

| | | | | | | | |
|--|---|-----|----|----|-----|---|-------------------------|
| Тема 6. Основы ООП | Содержание учебного материала Лекционные занятия Понятие об объектно-ориентированном подходе к программированию. Инкапсуляция, наследование, полиморфизм. Перечисления и структуры. Практические занятия Решение задач. Самостоятельная работа изучение теоретического материал | 39 | 11 | 11 | 17 | 2 | ПК 2.2, ПК 2.3 |
| Тема 7. Классы. | Лекционные занятия Понятие о классах. Поля, свойства, методы, индекаторы, конструкторы и деструкторы. Практические занятия Решение задач с использованием классов. Самостоятельная работа изучение теоретического материала. | 39 | 11 | 11 | 17 | | ПК 2.2, ПК 2.3 |
| Тема 8. Агрегатирование и композиция.. | Лекционные занятия Агрегатирование и композиция. Динамические структуры данных Практические занятия Решение задач: агрегирования и композиции. Самостоятельная работа изучение теоретического материала. | 39 | 11 | 11 | 17 | | ПК 2.2, ПК 2.3 |
| Тема 9. Наследование и полиморфизм. Интерфейсы. | Лекционные занятия Наследование и полиморфизм. Интерфейсы. Практические занятия Решение задач: наследования и полиморфизма. Самостоятельная работа изучение теоретического материала. | 39 | 11 | 11 | 17 | | ПК 2.2, ПК 2.3 |
| Тема 10. Делегаты и события. Коллекции. Универсальные типы. | Лекционные занятия Делегаты, события. Необобщенные коллекции. Обобщенные типы. Универсальные типы и интерфейсы. Практические занятия Решение задач : коллекции и универсальные типы данных. Самостоятельная работа изучение теоретического материала. | 40 | 12 | 11 | 17 | | ПК 2.2, ПК 2.3 |
| Всего | | 286 | 56 | 55 | 175 | | |

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – **ознакомительный или минимальный уровень** (узнавание ранее изученных объектов, свойств);
2. – **репродуктивный или базовый уровень** (выполнение деятельности по образцу, инструкции или под руководством)
3. – **продуктивный или высокий уровень (планирование и самостоятельное выполнение деятельности, решение проблемных задач)**

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Требования к минимальному материально-техническому обеспечению

Для реализации программы учебной дисциплины предусмотрены оборудованные помещения.

Основное оборудование учебной аудитории для лекционных занятий:

- рабочие места обучающихся;
- рабочее место преподавателя;
- маркерная (или меловая) доска.
- мультимедийное оборудование.

Программное обеспечение:

- ОС Microsoft Windows;
- Пакет приложений Microsoft Office (Open Office, Libre Office).

Основное оборудование учебной аудитории для практических (лабораторных) занятий:

- рабочие места обучающихся;
- автоматизированные рабочие места обучающихся;
- рабочее место преподавателя;
- маркерная (или меловая) доска.
- мультимедийное оборудование.

Программное обеспечение:

- ОС Microsoft Windows;
- Пакет приложений Microsoft Office (Open Office, Libre Office).
- Microsoft Visual Studio;
- СУБД: SQL Server, MySQL, PostgreSQL;
- Notepad++;
- Git;
- Microsoft Visio (DIA).

3.2. Информационное обеспечение реализации программы

Основная литература

1. Информатика / Е.Н. Гусева, И.Ю. Ефимова, Р.И. Коробков и др. – 4-е изд., стер. – Москва : Издательство «Флинта», 2016. – 261 с. : ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=83542>
2. Лубашева, Т.В. Основы алгоритмизации и программирования : [12+] / Т.В. Лубашева, Б.А. Железко. – Минск : РИПО, 2016. – 378 с. : ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=463632>
3. Борисенко, В. В. Основы программирования [Электронный ресурс] / В. В. Борисенко. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 323 с. — 978-5-9556-00039-0. — Режим доступа: <http://www.iprbookshop.ru/52206.html>
4. Волкова, Т.И. Введение в программирование / Т.И. Волкова. – Москва ; Берлин : Директ-Медиа, 2018. – 139 с. : ил., схем., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=493677>

Дополнительная литература

5. Хиценко, В.П. Основы программирования / В.П. Хиценко ; Министерство образования и науки Российской Федерации, Новосибирский государственный технический университет. – Новосибирск : НГТУ, 2015. – 83 с. : схем. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=438365>
6. Зыков, С. В. Основы современного программирования. Разработка гетерогенных систем в Интернет-ориентированной среде [Электронный ресурс] : учебный курс / С. В. Зыков. — Электрон. текстовые данные. — Саратов : Ай Пи Эр Медиа, 2017. — 484 с. —

978-5-9908055-9-0. — Режим доступа: <http://www.iprbookshop.ru/62072.html>

7. Колокольникова, А.И. Спецразделы информатики: основы алгоритмизации и программирования / А.И. Колокольникова. – Москва ; Берлин : Директ-Медиа, 2019. – 424 с. : ил., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=560695>

8. Тюгашев, А. А. Основы программирования. Часть 1 [Электронный ресурс] : учебное пособие / А. А. Тюгашев. — Электрон. текстовые данные. — СПб. : Университет ИТМО, 2016. — 163 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/67495.html>

9. Тюгашев, А. А. Основы программирования. Часть 2 [Электронный ресурс] : учебное пособие / А. А. Тюгашев. — Электрон. текстовые данные. — СПб. : Университет ИТМО, 2016. — 120 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/67496.html>

Ресурсы информационно-телекоммуникационной сети "Интернет", необходимые для освоения дисциплины (модуля)

| № п/п | Интернет ресурс (адрес) | Описание ресурса |
|-------|---|---|
| 1. | http://citforum.ru/ | IT-портал «Сервер Информационных Технологий»; |
| 2. | https://habrahabr.ru/ | ресурс для IT-специалистов |
| 3. | http://stackoverflow.com/ | сайт вопросов и ответов для IT-специалистов; |
| 4. | http://www.firststeps.ru | Первые шаги – Сайт, посвященный начинающим программистам. Учебники и инструкции для по языкам программирования, алгоритмам и используемым протоколам. Вопросы безопасности. |
| 5. | http://www.proklondike.com | Programmer'sKlondike - Бесплатная электронная библиотека. Книги по алгоритмам и дискретной математике. Учебники и статьи. |
| 6. | http://www.intuit.ru | Интернет-университет информационных технологий (ИНТУИТ) |
| 7. | https://msdn.microsoft.com/ru-ru/ | MSDN – сеть разработчиков Microsoft |
| 8. | https://mva.microsoft.com/ | Виртуальная академия Microsoft |

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

| <i>Результаты обучения, подлежащие проверке</i> | <i>Критерии оценки</i> | <i>Формы и методы оценки</i> |
|--|--|---|
| <i>Перечень знаний, осваиваемых в рамках дисциплины:</i> 31 основы структурного и объектно-ориентированного подхода к разработке программ, 32 об инкапсуляции, наследовании и полиморфизме, а также событийно-управляемых приложениях, | «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко. «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками. | Письменный и устный опросы Тестирование Оценка выполнения практического задания |

| | | |
|---|---|--|
| <p>33 назначение, основные свойства, методы и события стандартных элементов управления. <i>Перечень умений, осваиваемых в рамках дисциплины:</i> У1 использовать интегрированные среды разработки, языки программирования высокого уровня C# /Python и библиотеки классов для создания, тестирования и отладки консольных и оконных приложений.</p> | <p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки. «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.</p> | |
| <p>В результате освоения дисциплины обучающийся должен освоить следующие компетенции: ПК 2.2. Разрабатывать и публиковать программное обеспечение и информационные ресурсы отраслевой направленности со статическим и динамическим контентом на основе готовых спецификаций и стандартов. ПК 2.3. Проводить отладку и тестирование программного обеспечения отраслевой направленности.</p> | | |

Описание показателей и критериев оценивания компетенций

| Показатели оценивания | Критерии оценивания компетенций | Шкала оценивания |
|--|---|--|
| <p>Понимание смысла компетенции</p> | <p>Имеет базовые общие знания в рамках диапазона выделенных задач (1 балл) Понимает факты, принципы, процессы, общие понятия в пределах области исследования. В большинстве случаев способен выявить достоверные источники информации, обработать, анализировать информацию. (2 балла) Имеет фактические и теоретические знания в пределах области исследования с пониманием границ применимости (3 балла)</p> | <p>Минимальный уровень Базовый уровень Высокий уровень</p> |
| <p>Освоение компетенции в рамках изучения учебной дисциплины</p> | <p>Наличие основных умений, требуемых для выполнения простых задач. Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче (1 балл) Имеет диапазон практических умений, требуемых для решения определенных проблем в области исследования. В большинстве случаев способен выявить достоверные источники информации, обработать, анализировать информацию. (2 балла) Имеет широкий диапазон практических умений, требуемых для развития творческих решений, абстрагирования проблем. Способен выявлять</p> | <p>Минимальный уровень Базовый уровень</p> |

| | | |
|---|---|--|
| | проблемы и умеет находить способы решения, применяя современные методы и технологии. (3 балла) | Высокий уровень |
| Способность применять на практике знания, полученные в ходе изучения дисциплины | Способен работать при прямом наблюдении. Способен применять теоретические знания к решению конкретных задач. (1 балл) Может взять на себя ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем. Затрудняется в решении сложных, неординарных проблем, не выделяет типичных ошибок и возможных сложностей при решении той или иной проблемы (2 балла) | Минимальный уровень Базовый уровень |
| | Способен контролировать работу, проводить оценку, совершенствовать действия работы. Умеет выбрать эффективный прием решения задач по возникающим проблемам. (3 балла) | Высокий уровень |

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по выполнению лекционных занятий

Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии.

Методические указания по выполнению практических занятий

Проработка рабочей программы, уделяя особое внимание целям и задачам структуре и содержанию дисциплины. Конспектирование источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с текстом (указать текст из источника и др.). Прослушивание аудио- и видеозаписей по заданной теме, решение расчетно-графических заданий, решение задач по алгоритму и др.

Методические указания по выполнению лабораторных работ/индивидуальных заданий

Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующих для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др.

Методические указания по подготовке к экзамену.

Наиболее ответственным этапом в обучении студентов является экзаменационная сессия. На ней студенты отчитываются о выполнении учебной программы, об уровне и объеме полученных знаний. Это государственная отчетность студентов за период обучения, за изучение учебной дисциплины.

Залогом успешного прохождения контроля являются систематические, добросовестные занятия студента. Однако это не исключает необходимости специальной работы перед сессией и в период сдачи зачета. Специфической задачей студента в период экзаменационной сессии являются повторение, обобщение и систематизация всего материала.

В процессе повторения анализируются и систематизируются все знания, накопленные при изучении программного материала: данные учебника, записи

лекций, конспекты прочитанных книг, заметки, сделанные во время консультаций или семинаров, и др.

Консультации, которые проводятся для студентов в период экзаменационной сессии, необходимо использовать для углубления знаний, для восполнения пробелов и для разрешения всех возникших трудностей.

При подготовке к контролю необходимо еще раз проверить себя на предмет усвоения основных категорий и ключевых понятий курса.

6. ОЦЕНОЧНЫЕ СРЕДСТВА И КОНТРОЛЬНО-ИЗМЕРИТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

6.1. Оценочные средства для проведения текущего контроля по учебной дисциплине

Тема 1. Алгоритмы (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Свойства и виды алгоритмов.
2. Исполнители алгоритмов (назначение, среда, режим работы, система команд).
3. Компьютер как формальный исполнитель алгоритмов.
4. Блок-схемы, графические элементы.
5. Базовые алгоритмические конструкции: СЛЕДОВАНИЕ, РАЗВИЛКА, ПОВТОРЕНИЕ.
6. Использование псевдокода. Разработка алгоритмов путем пошаговой детализации.

Практические занятия:

Решение задач на составление блок-схем.

Тема 2. Системы счисления. Кодирование данных. (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Единицы измерения количества информации.
2. Непозиционные и позиционные системы счисления. 2-, 8- и 16-ричные системы счисления, перевод чисел из одной системы в другую.
3. Кодирование символов, таблицы символов ASCII, ANSI, Unicode.
4. Кодирование графической информации. Глубина цвета и количество отображаемых цветов. Цветовые модели RGB и CMYK.

Практические занятия:

1. Решение задач на перевод данных из одной системы счисления в другую.

Самостоятельная работа: изучение теоретического материала.

Тема 3. Структура программы на C# / Python. Ввод/вывод. Структуры управления (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Структура программы на C# / Python.
2. Простейший ввод/вывод.
3. Структуры управления: ветвящиеся и циклические алгоритмы и программы..

Практические занятия:

1. Решение задач на реализацию ветвящихся и циклических алгоритмов.

Самостоятельная работа: изучение теоретического материала.

Тема 4. Массивы (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Одномерные массивы
2. многомерные массивы

3. ступенчатые массивы

Практические занятия:

1. Решение задач по работе массивами

Самостоятельная работа: изучение теоретического материала.

Тема 5. Рекурсия (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Рекурсивные алгоритмы и программы.

Практические занятия:

1. Решение задач на реализацию рекурсивных алгоритмов.

Самостоятельная работа: изучение теоретического материала.

Тема 6. Основы ООП (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Понятие об объектно-ориентированном подходе к программированию.
2. Инкапсуляция, наследование, полиморфизм.
3. Перечисления и структуры.

Практические занятия (лабораторные работы):

1. Решение задач.

Самостоятельная работа: изучение теоретического материала.

Тема 7. Классы (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Понятие о классах.
2. Поля, свойства, методы, индексы
3. конструкторы и деструкторы.

Практические занятия (лабораторные работы):

1. Решение задач с использованием классов.

Самостоятельная работа: изучение теоретического материала.

Тема 8. Агрегатирование и композиция (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Агрегатирование и композиция.
2. Динамические структуры данных

Практические занятия (лабораторные работы):

1. Решение задач: агрегирования и композиции.

Самостоятельная работа: изучение теоретического материала.

Тема 9. Наследование и полиморфизм. Интерфейсы (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Наследование и полиморфизм.
2. Интерфейсы.

Практические занятия (лабораторные работы):

1. Решение задач: наследования и полиморфизма.

Самостоятельная работа: изучение теоретического материала.

Тема 10. Делегаты и события. Коллекции. Универсальные типы (ПК 2.2, ПК 2.3)

Вопросы к обсуждению:

1. Делегаты, события.
2. Необобщенные коллекции.
3. Обобщенные типы.
4. Универсальные типы и интерфейсы

Практические занятия (лабораторные работы):

1. Решение задач : коллекции и универсальные типы данных.

Самостоятельная работа: изучение теоретического материала.

6.2. Контрольно-измерительные материалы для проведения текущего контроля по учебной дисциплине

Шкала оценки для проведения текущего контроля по учебной дисциплине в устной форме

| № п/п | Оценка за ответ | Характеристика ответа |
|-------|-----------------|--|
| 1 | Отлично | <ul style="list-style-type: none"> - полно раскрыто содержание материала; - материал изложен грамотно, в определенной логической последовательности; - точно используется терминология; - показано умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации; - продемонстрированные знания и умения позволяют самостоятельно решать поставленные задачи; - ответ прозвучал самостоятельно, без наводящих вопросов; - продемонстрирована способность творчески применять знание теории к решению профессиональных задач; - допущены одна - две неточности при освещении второстепенных вопросов, которые исправляются по замечанию. - количество баллов за освоение компетенций от 8 до 9 |
| 2 | Хорошо | <ul style="list-style-type: none"> - вопросы излагаются систематизировано и последовательно; - продемонстрированные знания и умения позволяют самостоятельно решать поставленные задачи, однако требуют определенного контроля; - продемонстрировано умение анализировать материал, однако не все выводы носят аргументированный и доказательный характер; - ответ удовлетворяет в основном требованиям на оценку «отлично», но при этом имеет один из недостатков: в изложении допущены небольшие пробелы, не исказившие содержание ответа; приобретенный практический опыт, знания и умения требуют незначительной корректировки в процессе выполнения задания; допущены ошибка или более двух недочетов при освещении второстепенных вопросов, которые легко исправляются по замечанию преподавателя. |

| | | |
|---|---------------------|---|
| | | - количество баллов за освоение компетенций от 5 до 7 |
| 3 | Удовлетворительно | - неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения материала; - при неполном знании теоретического материала выявлен недостаточный уровень знаний и умений; студент не может применить теоретические знания на практике; - количество баллов за освоение компетенций от 3 до 4 |
| 4 | Неудовлетворительно | - не раскрыто основное содержание учебного материала; - обнаружено незнание или непонимание большей или наиболее важной части учебного материала; - допущены ошибки в определении понятий, при использовании терминологии, которые не исправлены после нескольких наводящих вопросов - отсутствуют практический опыт, знания и умения по предлагаемым ситуационным вопросам или задачам, количество баллов за освоение компетенций менее 3. - отказ от ответа или отсутствие ответа |

Шкала оценки для проведения текущего контроля по учебной дисциплине в письменной форме

| № п/п | Оценка за ответ | Характеристика ответа |
|-------|---------------------|---|
| 1 | Отлично | Материал раскрыт полностью, изложен логично, без существенных ошибок, выводы доказательны и опираются на теоретические знания Количество баллов за освоение материала от 8 до 9 |
| 2 | Хорошо | Основные положения раскрыты, но в изложении имеются незначительные ошибки выводы доказательны, но содержат отдельные неточности Количество баллов за освоение материала от 5 до 7 |
| 3 | Удовлетворительно | Изложение материала не систематизированное, выводы недостаточно доказательны, аргументация слабая. Количество баллов за освоение материала от 3 до 4 |
| 4 | Неудовлетворительно | Не раскрыто основное содержание материала, обнаружено незнание основных положений темы. Не сформированы компетенции, умения и навыки. Количество баллов за освоение компетенций менее 3 Ответ на вопрос отсутствует |

7. ОЦЕНОЧНЫЕ СРЕДСТВА И КОНТРОЛЬНО-ИЗМЕРИТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

7.1. Оценочные средства для проведения промежуточной аттестации по учебной дисциплине

Контрольные вопросы для подготовки к контролю при проведении промежуточной аттестации по учебной дисциплине:

1. Свойства и виды алгоритмов. Исполнители алгоритмов (назначение, среда, режим работы, система команд). Компьютер как формальный исполнитель алгоритмов.
2. Блок-схемы, графические элементы. Базовые алгоритмические конструкции: СЛЕДОВАНИЕ, РАЗВИЛКА, ПОВТОРЕНИЕ.
3. Использование псевдокода. Разработка алгоритмов путем пошаговой детализации.
4. Единицы измерения количества информации. Непозиционные и позиционные системы счисления. 2-, 8- и 16-ричные системы счисления, перевод чисел из одной системы в другую.
5. Кодирование символов, таблицы символов ASCII, ANSI, Unicode.
6. Кодирование графической информации. Глубина цвета и количество отображаемых цветов. Цветовые модели RGB и CMYK.
7. Структура программы на C#. Простейший ввод/вывод.
8. Структуры управления: ветвящиеся и циклические алгоритмы и программы.
9. Одномерные массивы: описание, создание и использование. Пример.
10. Многомерные массивы: описание, создание и использование. Пример.
11. Ступенчатые массивы: описание, создание и использование. Пример.
12. Рекурсивные алгоритмы и программы.
13. Понятие об объектно-ориентированном подходе к программированию. Инкапсуляция, наследование, полиморфизм.
14. Перечисления и структуры: описание, создание и использование. Пример.
15. Понятие о классах. Поля, свойства, методы.
16. Понятие о классах. Методы, конструкторы и деструкторы.
17. Понятие о классах. Индексаторы.
18. Агрегатирование и композиция.
19. Динамические структуры данных.
20. Наследование и полиморфизм.
21. Абстрактные классы и методы.
22. Интерфейсы.
23. Стандартные интерфейсы .Net
24. Интерфейсы: описание, реализация. Пример.
25. Делегаты: описание, создание и использование.
26. События: описание, создание и использование.
27. Необобщенные коллекции.
28. Универсальные типы. Понятие.
29. Универсальные типы и интерфейсы .Net.
30. Основы разработки Windows-приложений.
31. Программирование элементов управления. Свойства, методы и события. Метки, поля, флажки, кнопки.
32. Элементы управления. Свойства, методы и события. Списковые элементы.
33. События мыши и клавиатуры.
34. Создание меню и панелей инструментов.
35. Использование окон диалога.
36. Простой и многодокументный интерфейс. SDI-, MDI- и Explorer-style интерфейс.
37. Простой и многодокументный интерфейс. MDI-приложения.
38. Развертывание Windows-приложений.
39. Доступ к файлам. Простейший ввод/вывод.
40. Доступ к файловой системе. Диски, папки, файлы.

Итоговый тест (ПК 2.2, ПК 2.3)

1. Переменная – это:
 - a) объект, способный принимать различные значения;
 - b) значения чисел;

- c) меняющееся число;
- d) динамический объект.

2. Синтаксис языка программирования – это:

- a) набор правил расстановки знаков препинания;
- b) система правил, определяющая допустимые конструкции языка;
- c) интерпретация отдельных языковых конструкций языка;
- d) фиксированный набор основных символов, допускаемых для составления программы.

3. Функция в языке программирования – это:

- a) программный объект, принимающий значение с помощью оператора присваивания;
- b) программный объект, задающий вычислительную процедуру определения значения от аргумента;
- c) сегмент программы, хранящий некоторое значение, зависящее от аргумента;
- d) выражения, означающее зависимость левой части от правой.

4. Файл – это:

- a) база данных;
- b) поименованный участок на внешних носителях ;
- c) список;
- d) раздел на жестком диске.

5. Множество – это:

- a) список элементов, заключенный в круглые скобки, вида: <имя поля>:<значение>;
- b) неупорядоченная совокупность отличных друг от друга однотипных элементов;
- c) совокупность с фиксированным числом однотипных элементов, отличных только индексами;
- d) совокупность отличных друг от друга элементов разных типов.

6. Массив – это:

- a) запись множества переменных разного типа;
- b) неупорядоченная совокупность отличных друг от друга однотипных элементов;
- c) последовательность, состоящая из фиксированного числа однотипных элементов;
- d) тип одномерных величин.

7. Что будет выведено в результате выполнения программы?

```
view plaincopy to clipboardprint?  
class Program  
{  
    static void Main(string[] args)  
    {  
        var a = null;  
        a = 10;  
        Console.WriteLine(a);  
        Console.ReadLine();  
    }  
}
```


- a) 0
- b) 10
- c) Пустая строка
- d) Возникнет ошибка на этапе компиляции

8. Что будет выведено в результате выполнения программы?

```
view plaincopy to clipboardprint?
class Program
{
enum En
{
First = 15,
Second,
Third = 54
}
static void Main(string[] args)
{
Console.WriteLine((int)En.Second);
Console.ReadLine();
}
}
```

- a) 0
- b) 1
- c) 16
- d) Возникнет ошибка на этапе компиляции

9. Когда и почему использовать StringBuilder предпочтительнее чем string?

- a) Если строка редко изменяется
- b) Если строка часто изменяется
- c) Если строка содержит спецсимволы
- d) Если строка содержит исключительно цифры

10. К неявным преобразованиям относятся:

- a) преобразования, выполняемые автоматически
- b) любые преобразования над объектами ссылочных типов
- c) преобразования, указанные программистом

11. Закрытые методы класса

- a) составляют интерфейс класса
- b) не могут быть вызваны клиентами
- c) позволяют клиентам не вникать в детали реализации

12. Три массива: <int[] ar1 = new int[5]; double[] ar2 = {5.5, 6.6, 7.7}; int[,] ar3 = new Int32[3,4]; > принадлежат

- a) одному и тому же классу
- b) разным классам, общность которых определяется только родительским классом object
- c) разным классам, но у которых есть общий родительский класс Array

13. Правда ли, что:

- a) стандартные математические функции представлены статическими методами класса Math

- b) метод Next класса Random является перегруженным еще на этапе компиляции возникнет ошибка в следующем фрагменте кода `int x=1;int y=2;if (Math.Sin(x) > 0.5) y=1;`

14. При создании объектов перечислений

- a) объекты получают значение, даже если инициализация не задана
- b) требуется их явная инициализация
- c) нельзя вызывать конструктор

15. Отметьте истинное высказывание:

- a) каждая case-ветвь оператора switch должна заканчиваться оператором break
- b) любое выражение может использоваться в качестве оператора
- c) оператор foreach не применим при работе с массивами
- d) оператор if может быть эквивалентен по своему действию пустому оператору

16. При определении собственных классов можно задать:

- a) только конструкторы с атрибутом public
- b) конструкторы с атрибутами private и protected
- c) статический конструктор

17. Класс имеет:

- a) несколько конструкторов с одинаковой сигнатурой
- b) не более одного статического конструктора
- c) не более одного закрытого конструктора
- d) только конструктор по умолчанию
- e) только один конструктор

18. Отметьте истинные высказывания:

- a) класс Form1 в Windows-проекте по умолчанию является наследником класса Form
- b) проект библиотеки классов может не содержать точки входа Main
- c) пространство имен допускает вложенность и может содержать в качестве элемента пространство имен

19. Преобразование к типу object

- a) всегда возможно и выполняется автоматически
- b) возможно только для объектов ссылочного типа
- c) возможно, но требует явного задания преобразования

20. Для выделения из текста подстроки, в которой x и y разделены знаком операции + или -, следует использовать шаблон:

- a) `@>x[\\+|-]y<>`
- b) `@>x[+|-]y<>`
- c) `@>(x+y)|(x-y)<>`

21. Правда ли, что:

- a) под десериализацией понимают процесс восстановления состояния объектов, хранимых в долговременной памяти
- b) создать объект класса интерфейса обычным путем с использованием конструктора и операции new нельзя
- c) различают три типа клонирования: поверхностное (shallow), глубокое (deep) и сверхглубокое (overdeep)

22. Правда ли, что:
- a) при вызове конструктора класса ему может быть передана ссылка на создаваемый конструктором объект
 - b) поле класса может быть объектом описываемого класса
 - c) потомкам доступны все поля и методы родительского класса
23. Начальный Windows-проект содержит:
- a) класс Window
 - b) пространство имен Form
 - c) класс Form
 - d) класс Form1 – наследника класса Form
24. Класс Exception
- a) имеет фиксированное число потомков
 - b) является абстрактным классом
 - c) является родительским классом для всех классов, описывающих определенный тип исключения
25. Размерность массива определяется:
- a) числом его элементов
 - b) числом его индексов
 - c) по максимальному значению элементов
26. Правда ли, что:
- a) метод Format может вызываться с разным числом параметров
 - b) существуют константы класса StringBuilder
 - c) метод Join позволяет воссоздать строку в исходном виде, расщепленную методом Split
27. К значимым типам языка C# относятся:
- a) строки
 - b) все арифметические типы, кроме типа double
 - c) все арифметические типы
 - d) массивы
28. Для ограниченной универсальности справедливы следующие утверждения:
- a) ограничение универсальности накладывает ограничения на все родовые параметры
 - b) неограниченную универсальность ограничивает статический контроль типов
 - c) введение ограничений универсальности на родовой параметр T сужает возможности работы с объектами типа T
 - d) в одном предложении where можно задать ограничения на несколько родовых параметров
29. Отладка позволяет
- a) доказать, что в системе нет ошибок
 - b) устранить все ошибки, существующие в системе
 - c) доказать, что в системе есть ошибки
30. Оператор return
- a) завершает выполнение функции или процедуры
 - b) возможен в процедурах
 - c) обязателен в функциях

7.2. Контрольно-измерительные материалы для проведения промежуточной аттестации по учебной дисциплине
Шкала оценки для проведения промежуточной аттестации по учебной дисциплине в устной форме

| № п/п | Оценка за ответ | Характеристика ответа |
|-------|-------------------|--|
| 1 | Отлично | <ul style="list-style-type: none"> - полно раскрыто содержание материала; - материал изложен грамотно, в определенной логической последовательности; - точно используется терминология; - показано умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации; - продемонстрированные знания и умения позволяют самостоятельно решать поставленные задачи; - ответ прозвучал самостоятельно, без наводящих вопросов; - продемонстрирована способность творчески применять знание теории к решению профессиональных задач; - допущены одна - две неточности при освещении второстепенных вопросов, которые исправляются по замечанию. - количество баллов за освоение компетенций от 8 до 9 |
| 2 | Хорошо | <ul style="list-style-type: none"> - вопросы излагаются систематизировано и последовательно; - продемонстрированные знания и умения позволяют самостоятельно решать поставленные задачи, однако требуют определенного контроля; - продемонстрировано умение анализировать материал, однако не все выводы носят аргументированный и доказательный характер; - ответ удовлетворяет в основном требованиям на оценку «отлично», но при этом имеет один из недостатков: в изложении допущены небольшие пробелы, не искавшие содержание ответа; приобретенный практический опыт, знания и умения требуют не значительной корректировки в процессе выполнения задания; допущены ошибка или более двух недочетов при освещении второстепенных вопросов, которые легко исправляются по замечанию преподавателя. - количество баллов за освоение компетенций от 5 до 7 |
| 3 | Удовлетворительно | <ul style="list-style-type: none"> - неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения материала; - при неполном знании теоретического материала выявлен недостаточный уровень знаний и умений; студент не может применить теоретические знания на практике; - количество баллов за освоение компетенций от 3 до 4 |

| | | |
|---|---------------------|---|
| 4 | Неудовлетворительно | <ul style="list-style-type: none"> - не раскрыто основное содержание учебного материала; - обнаружено незнание или непонимание большей или наиболее важной части учебного материала; - допущены ошибки в определении понятий, при использовании терминологии, которые не исправлены после нескольких наводящих вопросов - отсутствуют практический опыт, знания и умения по предлагаемым ситуационным вопросам или задачам, количество баллов за освоение компетенций менее 3. - отказ от ответа или отсутствие ответа |
|---|---------------------|---|

Шкала оценки для проведения промежуточной аттестации по учебной дисциплине в письменной форме

| № п/п | Оценка за ответ | Характеристика ответа |
|-------|---------------------|---|
| 1 | Отлично | Материал раскрыт полностью, изложен логично, без существенных ошибок, выводы доказательны и опираются на теоретические знания Количество баллов за освоение материала от 8 до 9 |
| 2 | Хорошо | Основные положения раскрыты, но в изложении имеются незначительные ошибки выводы доказательны, но содержат отдельные неточности Количество баллов за освоение материала от 5 до 7 |
| 3 | Удовлетворительно | Изложение материала не систематизированное, выводы недостаточно доказательны, аргументация слабая. Количество баллов за освоение материала от 3 до 4 |
| 4 | Неудовлетворительно | Не раскрыто основное содержание материала, обнаружено незнание основных положений темы. Не сформированы компетенции, умения и навыки. Количество баллов за освоение компетенций менее 3 Ответ на вопрос отсутствует |

Критерии формирования оценок по тестам

| Оценка | Требования к знаниям |
|---------------------|----------------------|
| отлично | 80%-100% |
| хорошо | 65-80% |
| удовлетворительно | 50-65% |
| неудовлетворительно | менее 50% |
| зачтено | 50% и более |
| не зачтено | менее 50% |

8. ПРОВЕРКА СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Комплексное задание, проверяющее освоение группы компетенций ПК 2.2., ПК 2.3.

Комплексное задание

Вариант 1

Задание 1.

Можно ли из букв слова а составить слово б, причем каждую букву можно использовать только один раз?

Входные данные: Слово *a* в первой строке, *b* - во второй, состоящих из английских букв.

Выходные данные: Вывести *Ok* при положительном ответе и *No* в противном случае.

Задание 2.

Выведите квадраты всех натуральных чисел не больших *n* в возрастающем порядке.

Входные данные: Одно натуральное число *n* ($n \leq 109$).

Выходные данные: Выведите список квадратов всех натуральных чисел не больших *n* в возрастающем порядке.

Задание 3.

Задан массив из *n* целых чисел. Вывести только его положительные элементы, не меняя первоначальный порядок.

Входные данные: Первая строка содержит число *n* ($1 \leq n \leq 100$). Во второй строке записаны *n* целых чисел, каждое из которых не превышает по модулю 100.

Выходные данные: В первой строке вывести количество положительных элементов массива. Во второй строке вывести сами положительные элементы. Если положительных элементов в массиве нет, то вывести "NO".

Задание 4.

Задана последовательность целых чисел. Подсчитать количество элементов, у которых четные соседи.

Входные данные: В первой строке задано количество элементов последовательности *n* ($n \leq 100$). Во второй строке заданы сами элементы, значение каждого из которых по модулю не превышает 100.

Выходные данные: Вывести в одной строке количество элементов последовательности с четными соседями.

Задание 5.

Создать перечисление Цвет, классы Автомобиль и Автосалон.

Автомобиль – поля (марка, модель, цвет, цена), свойства, конструктор, переопределить подходящий метод для формирования информации об автомобиле.

Автосалон – поля (общедоступное: наименование, закрытый массив: автомобили), конструктор (с одним целочисленным параметром – число автомобилей в салоне), индексатор (по индексу возвращающий автомобиль – элемент массива автомобили), переопределить метод ToString(), чтобы он возвращал информацию обо всех автомобилях в салоне.

Вариант 2

Задание 1.

Задан массив из *n* целых чисел. Замените все наибольшие его элементы на наименьший, а наименьшие элементы на наибольший.

Входные данные: В первой строке записано число *n* ($n \leq 100$). В следующей строке записано *n* целых чисел, каждое из которых по модулю не превосходит 100.

Выходные данные: Вывести обновленный массив.

Задание 2.

Задана матрица *K*, содержащая *n* строк и *m* столбцов. Седловой точкой этой матрицы назовём элемент, который одновременно является минимумом в своей строке и максимумом в своём столбце.

Найдите количество седловых точек заданной матрицы.

Входные данные: Первая строка содержит целые числа *n* и *m*. ($1 \leq n, m \leq 750$). Далее

следуют n строк по m чисел в каждой. j -ое число i -ой строки равно $k_{i,j}$. Все $k_{i,j}$ по модулю не превосходят 1000.

Выходные данные: Выведите количество седловых точек.

Задание 3.

Задан массив из n целых чисел. Вывести пары соседних элементов одного знака. Если соседних элементов одного знака нет - ничего не выводите.

Входные данные: В первой строке записано число n . В следующей строке записано n целых чисел. Все числа по модулю не превышают 100.

Выходные данные: Вывести пары соседних элементов одного знака.

Задание 4.

Задан массив из n целых чисел. Найти сумму всех элементов массива, которые не равны максимальному.

Входные данные: В первой строке записано число n ($n \leq 100$). В следующей строке записано n целых чисел, каждое из которых не превышает по модулю 100.

Выходные данные: Вывести сумму всех элементов массива, которые не равны максимальному.

Задание 5.

Протестировать классы, построив объекты, описывающие два автосалона «Восток» и «Запад»:

- по два (Тойота, RAV4, белый, 1 млн.руб.; Хонда, CRV, черный, 1.2 млн.руб.) и
- три (Ситроен, С4, синий, 800 тыс.руб.; Форд, Фокус, серебристый, 750 тыс.руб.; Фольксваген, Тигуан, серый, 1 млн.руб.) автомобиля соответственно.

Построить диаграмму классов. Все напечатать.

ЛИСТ СОГЛАСОВАНИЯ

| №п/п | Подразделение | Фамилия | Подпись | Дата |
|------|--------------------|-----------------|---------|------------|
| 1 | Кафедра ГЕиМД | И.О. Тимофеева | | 10.06.2020 |
| 2 | Учеб.-метод. отдел | М.О. Дерябичева | | 10.06.2020 |
| 3 | Библиотека | Г.В. Шпакова | | 10.06.2020 |
| | | | | |