

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Краюшкина Марина Викторовна  
Должность: Директор  
Дата подписания: 13.12.2024 14:16:03  
Уникальный программный ключ:  
5e608be07b9761c0a5e2f0e4ccddb2e4db1e603

**Автономная некоммерческая организация профессионального образования  
«Университетский колледж»  
(АНО ПО «Университетский колледж»)**



**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ  
профессионального модуля**

**ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ**

по специальности среднего профессионального образования  
**09.02.07 Информационные системы и программирование**  
(код, наименование специальности)

Москва, 2024

Составлены в соответствии с Федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного Приказом Минпросвещения России от 9 декабря 2016 года № 1547.

Шифр дисциплины по стандарту – ПМ.01

## СОДЕРЖАНИЕ:

1. ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ
2. ПЛАНИРОВАНИЕ ВНЕАУДИТОРНОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ
3. КОНТРОЛЬ РЕЗУЛЬТАТОВ ВНЕАУДИТОРНОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ
4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МОДУЛЯ

## 1. ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа учащихся может рассматриваться как организационная форма обучения - система педагогических условий, обеспечивающих управление учебной деятельностью или деятельность учащихся по освоению общих и профессиональных компетенций, знаний и умений учебной и научной деятельности без посторонней помощи.

В учебном процессе выделяют два вида самостоятельной работы: аудиторная, внеаудиторная.

**Аудиторная самостоятельная работа** по учебной дисциплине и профессиональному модулю выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию.

**Внеаудиторная самостоятельная работа** выполняется учащимся по заданию преподавателя, но без его непосредственного участия.

*Самостоятельная работа учащихся проводится с целью:*

- систематизации и закрепления полученных теоретических знаний и практических умений студентов;
- углубления и расширения теоретических знаний;
- формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развития познавательных способностей и активности учащихся: творческой инициативы, самостоятельности, ответственности и организованности;
- формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развития исследовательских умений;
- формирования общих и профессиональных компетенций.

## 2. ПЛАНИРОВАНИЕ ВНЕАУДИТОРНОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Преподавателем учебной дисциплины эмпирически определяются затраты времени на самостоятельное выполнение конкретного содержания учебного задания: на основании наблюдений за выполнением учащимися аудиторной самостоятельной работы, опроса студентов о затратах времени на то или иное задание, хронометража собственных затрат на решение той или иной задачи с внесением поправочного коэффициента из расчета уровня знаний и умений учащихся.

При разработке рабочей программы по учебной дисциплине или профессиональному модулю при планировании содержания внеаудиторной самостоятельной работы преподавателей устанавливается содержание и объем теоретической учебной информации или практических заданий, которые выносятся на внеаудиторную самостоятельную работу, определяются формы и методы контроля результатов.

Содержание внеаудиторной самостоятельной работы определяется в соответствии с рекомендуемыми видами заданий согласно примерной программы учебной дисциплины или профессионального модуля.

**Видами заданий для внеаудиторной самостоятельной работы могут быть:**

- *для овладения знаниями:* компетентностно-ориентированные задание, чтение текста (учебника, первоисточника, дополнительной литературы): составление плана текста; графическое изображение структуры текста; конспектирование текста; реферирование текста; выписки из текста; работа со словарями и справочниками, ознакомление с нормативными документами; учебно-исследовательская работа; использование аудио- и видеозаписей, компьютерной техники и Интернета и др.;

- *для закрепления и систематизации знаний:* компетентностно-ориентированное задание, работа с конспектом лекции (обработка текста); повторная работа над учебным материалом (учебника, первоисточника, дополнительной литературы, аудио- и видеозаписей); составление плана и тезисов ответа; составление таблиц для систематизации учебного материала; изучение нормативных материалов; ответы на контрольные вопросы; аналитическая обработка текста (аннотирование, рецензирование, реферирование, контент-анализ и др.); подготовка сообщений к выступлению на семинаре, конференции; подготовка рефератов, докладов; составление библиографии, тематических кроссвордов; тестирование и др.;

- *для формирования компетенций:* компетентностно-ориентированное задание, решение задач и упражнений по образцу; решение вариативных задачи упражнений; выполнение чертежей, схем; выполнение расчетно-графических работ; решение ситуационных педагогических задач; подготовка к деловым играм; проектирование и моделирование разных видов и компонентов профессиональной деятельности; подготовка курсовых работ; опытно-экспериментальная работа; упражнения на тренажере; упражнения спортивно-оздоровительного характера; рефлексивный анализ профессиональных умений с использованием аудио- и видеотехники и др.

Виды заданий для внеаудиторной самостоятельной работы, их содержание и характер могут иметь вариативный и дифференцированный характер, учитывать специфику специальности, изучаемой дисциплины, индивидуальные особенности студента.

При предъявлении видов заданий на внеаудиторную самостоятельную работу рекомендуется использовать дифференцированный подход к студентам. Перед выполнением студентами внеаудиторной самостоятельной работы преподаватель проводит инструктаж по выполнению задания, который включает цель задания, его содержание, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. В процессе инструктажа преподаватель предупреждает учащихся о возможных типичных ошибках, встречающихся при выполнении задания.

Инструктаж проводится преподавателем за счет объема времени,

отведенного на изучение дисциплины.

Самостоятельная работа может осуществляться индивидуально или группами учащихся в зависимости от цели, объема, конкретной тематики самостоятельной работы, уровня сложности уровня умений учащихся.

Отчет по самостоятельной работе учащихся может осуществляться как в печатном, так и в электронном виде (на CD диске).

### **3. КОНТРОЛЬ РЕЗУЛЬТАТОВ ВНЕАУДИТОРНОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ**

Контроль результатов внеаудиторной самостоятельной работы студентов может осуществляться в пределах времени, отведенного на обязательные учебные занятия по дисциплине и внеаудиторную самостоятельную работу учащихся по дисциплине, может проходить в письменной, устной или смешанной форме, с представлением продукта деятельности учащегося.

В качестве форм и методов контроля внеаудиторной самостоятельной работы учащихся могут быть использованы, *зачеты, тестирование, самоотчеты, контрольные работы, защита творческих работ и др., которые могут осуществляться на учебном занятии или вне его (например, оценки за реферат).*

Критериями оценки результатов внеаудиторной самостоятельной работы учащегося являются:

- уровень освоения учащимся учебного материала;
- умение учащегося использовать теоретические знания при выполнении практических задач;
- сформированность общих и профессиональных компетенций;
- обоснованность и четкость изложения ответа;
- оформление материала в соответствии с требованиями.

## **МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ**

### **СООБЩЕНИЕ**

По содержанию сообщение может быть информационным или методическим. Информационное сообщение – это теоретические материалы по определенной теме, расширяющие знания в области психологии, педагогики, других дисциплин. Методическое сообщение отражает практико-ориентированную информацию о различных инновационных, эффективных, нестандартных, результативных аспектах конкретной дисциплины.

### **РЕФЕРАТ**

Реферат (от латинского — сообщаю) — краткое изложение в письменном виде содержания научного труда (трудов), литературы по теме. Это самостоятельная научно-исследовательская работа, где раскрывается суть исследуемой проблемы, изложение материала носит проблемно-тематический

характер, показываются различные точки зрения, а также собственные взгляды на проблему. Содержание реферата должно быть логичным.

#### **Критерии оценки реферата:**

- соответствие теме;
- глубина проработки материала;
- правильность и полнота использования источников;
- оформление реферата.

### **ДОКЛАД**

Доклад — вид самостоятельной работы учащихся, используется в учебных и внеклассных занятиях, способствует формированию навыков исследовательской работы, расширяет познавательные интересы, приучает практически мыслить. При написании доклада по заданной теме следует составить план, подобрать основные источники. Работая с источниками, попытаться систематизировать полученные сведения, сделать выводы и обобщения. В настоящее время в учебных заведениях доклады содержательно практически ничем не отличаются от рефератов. Структура и оформление доклада такое же, как в реферате.

#### **Оформление титульного листа методической работы**

На титульном листе посередине его записывается вид работы, ниже на 10 мм – её название строчными буквами, справа в нижнем углу - фамилия автора разработки, группа. В нижней части титульного листа посередине указывается год написания разработки.

### **Темы самостоятельной работы**

№ раздела (темы)	Вопросы, выносимые на самостоятельное изучение	Количество часов
		ОФО
МДК.01.01 Разработка программных модулей		
1.	Жизненный цикл ПО	16
2.	Структурное программирование	16
3.	Объектно-ориентированное программирование	16
4.	Паттерны проектирования	16
5.	Событийно- управляемое программирование	16
6.	Оптимизация и рефакторинг кода	18
7.	Разработка пользовательского интерфейса	18
8.	Основы ADO.Net	18
МДК.01.02 Поддержка и тестирование программных модулей		
1.	Отладка и тестирование программного обеспечения	36
2.	Документирование	36
МДК.01.03 Разработка мобильных приложений		
1.	Основные платформы и языки разработки мобильных приложений	32
2.	Создание и тестирование модулей для мобильных приложений	32
МДК.01.04 Системное программирование		
1.	Программирование на языке низкого уровня	72
<b>Всего</b>		<b>348</b>

## **Практические задания МДК.01.01 Разработка программных модулей**

1. Создать форму для ввода массива и кнопкой «Перемешать». При нажатии на кнопку элементы массива перемешиваются
2. Палиндромом называют последовательность символов, которая читается как слева направо, так и справа налево. Создать web-форму для ввода строки и кнопкой, при нажатии на которую определяется, является ли введенная строка палиндромом
3. Создать web-форму с двумя полями для ввода чисел и выбором операции (+, -, \*, /). Вычислить соответствующее выражение и вывести результат в браузер
4. Найти наибольшее из трех чисел, введенных пользователем в форму
5. Написать программу нахождения синуса и косинуса угла, выбор действия осуществляется с помощью переключателя
6. Средствами PHP вывести время и дату на странице
7. Создать web-форму для ввода строки и кнопкой, при нажатии на которую строка переворачивается и выводится в браузер
8. Написать программу, которая для введенной строки подсчитывает сумму всех содержащихся в ней цифр
9. Создать класс Point, разработав следующие элементы класса: поля (x, y); конструкторы, позволяющие создать экземпляр класса с нулевыми координатами или с заданными координатами; методы, позволяющие вывести координаты точки на экран, рассчитать расстояние от начала координат до точки, переместить точку на плоскости на вектор (a, b); свойства для получения-установки координаты точки (доступное для чтения и записи), для умножения координаты точки на скаляр (доступное только для записи)
10. Создать класс Triangle, разработав следующие элементы класса: поля (a, b, c); конструктор, позволяющий создать экземпляр класса с заданными длинами сторон; методы, позволяющие вывести длины сторон треугольника на экран, рассчитать периметр треугольника, рассчитать площадь треугольника; свойства для получения-установки длин сторон треугольника (доступное для чтения и записи), для определения, существует ли треугольник с данными длинами сторон (доступное только для чтения)
11. Создать класс Rectangle, разработав следующие элементы класса: поля (a, b); конструктор, позволяющий создать экземпляр класса с заданными длинами сторон; методы, позволяющие вывести длины сторон прямоугольника на экран, рассчитать периметр прямоугольника, рассчитать площадь прямоугольника; свойства для получения-установки длин сторон прямоугольника (доступное для чтения и записи), для определения, является ли данный прямоугольник квадратом (доступное только для чтения)
12. Создать класс Round, разработав следующие элементы класса: поля (r); конструктор, позволяющий создать экземпляр класса с заданным радиусом; методы, позволяющие рассчитать длину окружности, рассчитать площадь круга; свойства для получения-установки радиуса круга (доступное для чтения и записи)
13. Создать абстрактный класс Figure с методами вычисления площади и периметра, а также методом, выводящим информацию о фигуре на экран.



Создать производные классы: Triangle (треугольник), Circle (круг) со своими методами вычисления площади и периметра. Создать массив n фигур и вывести полную информацию о фигурах на экран

14. Посчитать сумму элементов побочной диагонали квадратной матрицы
15. Найти среднее арифметическое положительных элементов одномерного массива. Количество элементов массива вводится пользователем, массив заполняется случайными числами
16. Посчитать в строке количество цифр. Строка вводится пользователем с клавиатуры
17. Создать форму Windows. Программа должна предоставлять возможность менять цвет фона формы
18. Создать Windows-приложение, демонстрирующее работу с модальными и немодальными окнами
19. Написать программу на языке C++, которая считает сумму элементов одномерного массива, не используя операцию индексирования [].
20. Написать программу на языке C++, демонстрирующую способы передачи параметров в функции
21. Написать рекурсивную функцию для нахождения факториала числа
22. Написать программу на языке C++ для подсчета суммы элементов одномерного массива. Размерность массива вводится пользователем с клавиатуры
23. Написать программу, демонстрирующую возможность создания односвязного списка и добавления элемента в список
24. Написать программу, демонстрирующую возможность создания стека и добавления элемента в любое место стека
25. Создать класс «Прямоугольник», который имеет два конструктора. Первый конструктор принимает в качестве параметров длину и ширину. Второй конструктор создает квадрат с заданной стороной
26. Написать программу, демонстрирующую работу с абстрактными классами
27. Создать функцию на языке C++, подсчитывающую сумму элементов массива. Массив передается как параметр в функцию
28. Написать программу для нахождения максимального элемента матрицы.
29. Написать программу для сортировки массива методом пузырька.
30. Написать программу для нахождения корней квадратного уравнения  $ax^2+bx+c=0$ . Коэффициенты a, b, c вводятся пользователем

Темы рефератов (докладов, презентаций)

1. История развития CASE-средств разработки программного обеспечения
2. Диаграммы развертывания
3. Методология разработки ПО RUP (Rational Unified Process)
4. Методология разработки ПО Crystal Clear
5. Среда dotNet (.NetFramework)
6. Измерение и оценка характеристик качества ПО
7. Роль стандартизации и сертификации в управлении качеством ПС
8. Виды тестирования ПО
9. Тестирование объектно ориентированных систем

10. Тестирование Веб-приложений
11. Принципы обеспечения безопасности ПО на различных стадиях его жизненного цикла: при планировании работ и проектном анализе ПО, в процессе его разработки, при приемо-сдаточных испытаниях, при эксплуатации ПО
12. Принципы криптографической защиты информации
13. Технологии аутентификации
14. Правовые методы защиты программных продуктов и баз данных
15. Сопровождение программного средства

### **Практические задания МДК.01.02 Поддержка и тестирование программных модулей**

1. Дан двухмерный массив  $5 \times 5$ . Найти сумму модулей отрицательных нечетных элементов. Сформулировать требования к программному продукту и выполнить анализ и тестирование программных требований в соответствии со свойствами качественных требований.
2. Дана матрица. Вывести на экран все четные строки, то есть с четными номерами, у которых первый элемент больше последнего. Сформулировать требования к программному продукту и разработать чек – лист.
3. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту и разработать чек – лист.
4. Дана целочисленная квадратная матрица. Определить: произведение элементов в тех строках, которые не содержат отрицательных элементов. Сформулировать требования к программному продукту и разработать тест – кейс.
5. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и разработать набор тест – кейсов.
6. Две строки матрицы назовем *похожими*, если совпадают множества чисел, встречающихся в этих строках. Найдите все пары похожих строк в заданной матрице  $m \times n$ . Сформулировать требования к программному продукту и разработать набор тест – кейсов.
7. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по структурным критериям.
8. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по методу «белого ящика».
9. Дана матрица. Вывести на экран все четные строки, то есть с четными номерами, у которых первый элемент больше последнего. Сформулировать

требования к программному продукту и выполнить тестирование программного продукта по методу «белого ящика».

10. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование по методу «белого ящика».

11. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование программного продукта по структурным критериям.

12. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по функциональным критериям.

13. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по методу «черного ящика».

14. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование по методу «черного ящика».

15. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$  вещественных чисел). Сформулировать требования к программному продукту и выполнить тестирование по методу «белого ящика».

16. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$  вещественных чисел). Сформулировать требования к программному продукту и выполнить модульное тестирование

17. Описать функцию  $NewStr(S)$ , удаляющую в строке  $S$  начальные и конечные пробелы. В основной программе ввод строки, обращение методу - функции и вывод результата. Сформулировать требования к программному продукту и выполнить `unit` – тестирование.

18. Описать функцию  $NewStr(S)$ , удаляющую в строке  $S$  начальные и конечные пробелы. В основной программе ввод строки, обращение методу - функции и вывод результата. Предусмотреть использование 2 –х форм. Сформулировать требования к программному продукту и выполнить интеграционное тестирование.

19. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$  вещественных чисел). Предусмотреть использование 2 –х форм.

Сформулировать требования к программному продукту и выполнить интеграционное тестирование.

20. Дано натуральное число  $n$  и последовательность из 5 чисел. Найти количество чисел, являющихся степенями пятерки. Определить функцию пользователя, позволяющую распознавать степень пятерки. В основной программе ввод чисел, обращение к функции, вычисление количества и вывод результата. Сформулировать требования к программному продукту и выполнить модульное тестирование.

Темы рефератов (докладов, презентаций)

1. Принцип построения тестового набора данных и составления отладочных заданий.
2. Методы отладки программного продукта
3. Методы ручного тестирования
4. Метод обратного прослеживания
5. Инструментальные средства отладки ПП
6. Модели жизненного цикла программного средства.
7. Разработка программного продукта с использованием подпрограммы-процедуры.
8. Модульное программирование.
9. Осуществление разработки кода программного модуля на современных языках программирования
10. Реализация процедур и функций работы с бинарным деревом.
11. Объектно-ориентированное проектирование.
12. Основы языка UML (Unified Modeling Language).
13. Структура программы на языке C++. Проект.
14. Стандартная библиотека функций языка C++.
15. Компиляция программы и сборка исполняемого модуля.
16. Виртуальные функции и абстрактные базовые классы.
17. Множественное наследование.
18. Ассоциативные массивы.
19. Объекты-функции и предикаты.

### **Практические задания МДК.01.03 Разработка мобильных приложений**

1. Требуется разработать приложение с графическим пользовательским интерфейсом, поддерживающее создание/редактирование/удаление/поиск заметок. Два варианта хранения заметок: А) в базе SQLite. Б) С использованием файловой системы.
2. Создать приложение с графическим пользовательским интерфейсом с функциями: Определение местоположения пользователя на карте Google Map; Определение скорости и направления движения пользователя; Масштабирование карты. Программа должна быть конфигурируемой. Настройки: Режим определения местоположения (через GPS либо по сотам); Включение/отключение режима поиска.
3. Разработать приложение-таймер с использованием датчика ориентации в виде песочных часов. Каждый раз для того чтобы активировать таймер,

необходимо перевернуть экран мобильного устройства вверх ногами. Используйте анимацию для показа «перетекающего песка» и переворота песочных часов. Для задания времени перетекания песка требуется разработать push-notification сервер. Через форму ввода на сервере можно отправлять на клиент (приложение-таймер) указанное время (числовой ввод).

4. Разработка программы для обмена мгновенными сообщениями. Требуется разработать приложение для обмена мгновенными сообщениями через Wi-Fi/Bluetooth. Поддерживаемые режимы: 1. Активный режим. Приложение занимает весь экран, содержит поля для отправки сообщений и список принятых сообщений. 2. Режим уведомлений. Приложение через уведомления показывает принятые сообщения.

5. Разработка мобильного сайта с адаптивным дизайном. Требуется разработать сайт, пригодный для просмотра на смартфонах и планшетах, с использованием принципов адаптивного дизайна: 1) Гибкая сетка. 2) Пропорциональные шрифты. 3) Масштабируемые изображения. 4) Медиазапросы

#### Темы рефератов (докладов, презентаций)

1. Программный стек Android. Виртуальная машина Dalvik.
2. Архитектура Android-приложений.
3. Четыре пункта философии разработки приложений под Android.
4. Приемы для улучшения производительности и уменьшения потребления памяти для приложений Android.
5. Основные составляющие манифеста приложения.
6. Жизненный цикл мобильного приложения.
7. Разработка интерфейсов, не зависящих от разрешения и плотности пикселей.
8. Intents и Activities. Принципы работы Intent-фильтров.
9. Адаптеры и привязка данных.
10. Работа с интернет-ресурсами.
11. Диалоговые окна: создание и использование.
12. Курсоры, Content Values. Получение данных из SQLite.
13. Фоновые службы, toast-уведомления и сигнализация.
14. Геолокационные и картографические сервисы: конфигурирование и использование.
15. Сенсорные датчики. Sensor manager.
16. Анимация и спецэффекты.
17. Акселерометр, датчик ориентации и компас: регулировка и программные функции.
18. Межпроцессное взаимодействие. Язык AIDL.
19. Основные права и полномочия для запуска приложений на устройстве.
20. Работа с настройками сотовой сети, подключение голосовых услуг, получение и отправка коротких сообщений.

#### **Практические задания МДК.01.04 Системное программирование**

1. Написать программу, которая удаляет из введенной с клавиатуры строки

первые пробелы (если их нет, то оставить как есть)

2. Подсчитать количество слов в строке
3. Выделить первое слово из строки
4. Найти максимальное из трех введенных целых чисел
5. Найти сумму цифр трехзначного числа
6. Решить квадратное уравнение  $ax^2+bx+c=0$ , коэффициенты уравнения вводятся с клавиатуры
7. Определить, кратна ли трем сумма цифр трехзначного числа. Число вводится с клавиатуры
8. Найти площадь прямоугольного треугольника по известным катету и гипотенузе. Исходные данные вводятся с клавиатуры
9. Переписать все символы строки в обратном порядке
10. Написать программу для сортировки одномерных массивов
11. Вывести на экран все трехзначные числа, в которых хотя бы две цифры одинаковые
12. Вывести на экран все трехзначные числа, которые начинаются и заканчиваются на одну и ту же цифру
13. Вывести на экран таблицу значений функции  $y=x^3$  при  $x=-4...4$ . Шаг изменения  $x$  вводится с клавиатуры
14. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 найти сумму положительных элементов
15. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 количество отрицательных элементов
16. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 найти среднее арифметическое четных элементов
17. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 заменить все отрицательные элементы нулями
18. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 переставить местами минимальный и максимальный элементы
19. Определить среднее арифметическое элементов главной диагонали квадратной матрицы
20. Определить максимальный элемент побочной диагонали квадратной матрицы
21. Найти сумму нечетных элементов прямоугольного массива, состоящего из пяти строк и семи столбцов
22. В прямоугольном массиве, состоящем из 7 строк и 10 столбцов заменить все отрицательные элементы их абсолютными значениями
23. Определить, является ли сумма элементов матрицы  $3 \times 2$  нечетной
24. В прямоугольном массиве, состоящем из 7 строк и 10 столбцов заменить все отрицательные элементы их абсолютными значениями
25. Определить, является ли сумма элементов матрицы  $3 \times 2$  нечетной

## Темы рефератов (докладов, презентаций)

1. Работа на ассемблере с портами ввода-вывода.
2. Визуальный анализ данных
3. Сравнительные характеристики автоматических дизассемблеров
4. Организация интерфейса с процедурой на ассемблере
5. Создание Windows – приложений на ассемблере
6. Программирование сопроцессора
7. Связь ассемблера с другими языками программирования (Pascal – ассемблер и С – ассемблер)
8. Сложные структуры данных в ассемблере (структуры, записи, списки, таблицы)
9. Работа с числами в формате BCD на ассемблере
10. Обработка цепочек элементов в ассемблере
11. Измерение скорости работы программы средствами профайлера
12. Построение лексического анализатора
13. Организация таблиц идентификаторов транслятора по методу бинарного дерева
14. Организация таблиц идентификаторов транслятора с использованием хеш-функций
15. Развитие отладчиков в современных системах программирования
16. Возможности Нех-редакторов
17. Преобразование чисел при вводе с консоли на ассемблере
18. Преобразование чисел при выводе на консоль на ассемблере
19. Управление цветом при выводе информации на консоль средствами ассемблера
20. Функции BIOS и MS DOS для работы с клавиатурой
21. Организация ввода-вывода в консольном приложении Windows
22. Программные средства обслуживания CD и DVD дисков
23. Сравнение средств разработки ассемблерных программ
24. Алгоритмы работы с памятью на уровне операционной системы (выделение, освобождение)
25. Использование концепции **.net** в системном программировании
26. Структура компьютерных технологий
27. Ошибки округления в арифметических командах
28. Использование флагов переноса и переполнения в командах целочисленной арифметики и передачи управления
29. Сравнительные характеристики трансляторов
30. Современные системы программирования
31. Организация серверных приложений (CORBA, COM/DCOM)

#### 4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МОДУЛЯ

##### *Основная литература:*

1. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 248 с. — (Профессиональное образование). — ISBN 978-5-534-18131-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539215> (дата обращения: 25.11.2024).

2. Трофимов, В. В. Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская. — 4-е изд. — Москва : Издательство Юрайт, 2024. — 108 с. — (Профессиональное образование). — ISBN 978-5-534-20429-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/558137> (дата обращения: 25.11.2024).

##### *Дополнительная литература:*

1. Рабчевский, А. Н. Компьютерные сети и системы связи. Вводный курс : учебное пособие для среднего профессионального образования / А. Н. Рабчевский. — Москва : Издательство Юрайт, 2024. — 226 с. — (Профессиональное образование). — ISBN 978-5-534-19073-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/555886> (дата обращения: 25.11.2024).

2. Акопов, А. С. Компьютерное моделирование : учебник и практикум для среднего профессионального образования / А. С. Акопов. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 426 с. — (Профессиональное образование). — ISBN 978-5-534-18369-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/534872> (дата обращения: 25.11.2024).

3. Системы управления технологическими процессами и информационные технологии : учебное пособие для среднего профессионального образования / В. В. Троценко, В. К. Федоров, А. И. Забудский, В. В. Комендантов. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 136 с. — (Профессиональное образование). — ISBN 978-5-534-09939-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539749> (дата обращения: 25.11.2024).



### ***Интернет-источники:***

- Огромный выбор конспектов лабораторных и практических работ, инструкционные и технологические карты почти на все основные темы курса <http://www.metod-kopilka.ru/page-2-1.html>
- Электронная библиотечная система (ЭБС) «Университетская библиотека ONLINE» <https://urait.ru/>
- <https://www.postgresql.org/about/licence>
- <https://www.componentsource.com/product/komodo-ide/licensing>
- <https://blog.jetbrains.com/pycharm/2017/09/pycharm-community-edition-and-professional-edition-explained-licenses-and-more>
- <https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html>
- <https://dbeaver.com/academic-license>
- <https://dbeaver.com/eula>
- <https://www.postgresql.org/about/licence>