

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Смирнов Сергей Николаевич
Должность: врио-ректора
Дата подписания: 18.10.2019 11:56
Уникальный программный ключ:
69e375c64f7e975d4e8830e7b4fcc2ad1bf35f08

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Тверской государственный университет»



Утверждаю:

Руководитель ООП

А.В. Язенин /

» июня 2019 года

Рабочая программа дисциплины (с аннотацией)

ПРАКТИКУМ НА ЭВМ

Направление подготовки

**02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА
И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

Профиль подготовки

Информатика и компьютерные науки

Для студентов 1-2-го курсов

Форма обучения – очная

Составитель:

д.ф.-м.н., доцент С.М. Дудаков

Л.Ф. Мальцева

Тверь, 2019

I. Аннотация

1. Цель и задачи дисциплины:

Целью освоения дисциплины является: выработать у студентов практические навыки по проектированию алгоритмов и их программной реализации на языках высокого уровня (C/C++). Задачами освоения дисциплины являются: развитие у студентов навыков алгоритмического мышления, разработка ими алгоритмов решения основных типовых задач включая лексический и синтаксический анализ текстов.

2. Место дисциплины в структуре ООП

Дисциплина входит в раздел «Информатика и коммуникационно-информационные технологии» обязательной части блока 1.

Предварительные знания и навыки. знание школьных курсов математики и информатики

Дальнейшее использование. Данный курс предназначен для сопровождения теоретического курса по программированию лабораторными занятиями. Полученные знания используются в дальнейшем при изучении других дисциплин, связанных с программированием, при прохождении практики, написании выпускной работы, а также в дальнейшей трудовой деятельности выпускников.

3. Объем дисциплины: 10 зач. ед., 360 акад. ч., в том числе:

контактная аудиторная работа практических занятий 109 ч., в том числе практическая подготовка 109 ч., лабораторных занятий 124 ч., в том числе практическая подготовка 124 ч.,

контактная внеаудиторная работа контроль самостоятельной работы 0 ч., в том числе курсовая (расчетно-графическая) работа 0 ч.;

самостоятельная работа 127 ч., в том числе контроль 0 ч.

4. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы:

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ОПК-2, Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отече-	ОПК-2.1, Знает основные положения и концепции в области программного обеспечения, теории коммуникации, знает основную терми-

<p>Планируемые результаты освоения образовательной программы (формируемые компетенции)</p>	<p>Планируемые результаты обучения по дисциплине</p>
<p>ственного происхождения, для решения задач профессиональной деятельности</p>	<p>нологию, знаком с основными классами программного обеспечения ОПК-2.2, Применяет программное обеспечение для решения типовых задач ОПК-2.3, Интегрирует различные типы программного обеспечения и коммуникации для решения профессиональных задач</p>
<p>ОПК-3, Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям</p>	<p>ОПК-3.1, Знает основные положения и концепции в области программирования ОПК-3.2, Знает архитектуру языков программирования ОПК-3.3, Составляет программы</p>

5. Форма промежуточной аттестации и семестр прохождения:

зачеты в 1–4 семестрах

6. Язык преподавания:

русский

II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Для студентов очной формы обучения

Учебная программа — наименование разделов и тем	Всего (час.)	Контактная работа (час.)					Сам. раб., в т.ч. контроль (час.)
		Лекции		Практ. занятия / Лаб. работы		Контроль сам. раб., в т.ч. курсовая работа	
		Всего	В т.ч. практ. подг.	Всего	В т.ч. практ. подг.		
1	2	3	4	5	6	7	8
Основные типы вычислительных процессов и примеры их алгоритмизации.	13	0	0	3/6	3/6	0	4
Базовые структуры данных. Массивы.	55	0	0	8/16	8/16	0	31
Функции.	24	0	0	4/8	4/8	0	12
Строки.	23	0	0	8/8	8/8	0	7
Структуры.	6	0	0	2/2	2/2	0	2
Файлы	27	0	0	10/10	10/10	0	7
Динамические структуры данных.	29	0	0	12/12	12/12	0	5
Классы.	23	0	0	6/6	6/6	0	11
Наследование.	11	0	0	4/4	4/4	0	3
Обработка исключительных ситуаций.	11	0	0	4/4	4/4	0	3
Стандартные потоки.	16	0	0	6/6	6/6	0	4
Файловые и строковые потоки.	12	0	0	4/4	4/4	0	4
Стандартная библиотека шаблонов STL.	17	0	0	6/6	6/6	0	5
Конечные автоматы.	6	0	0	2/2	2/2	0	2
Лексический блок для языка MINI-BASIC.	22	0	0	8/8	8/8	0	6
Автоматы с магазинной памятью.	6	0	0	2/2	2/2	0	2
Контекстно-свободные и транслирующие грамматики.	11	0	0	4/4	4/4	0	3
Синтаксический блок для языка MINI-BASIC.	22	0	0	8/8	8/8	0	6
Генератор кода для MINI-BASIC-компилятора.	26	0	0	8/8	8/8	0	10
Итого	360	0	0	109/124	109/124	0	127

Учебная программа дисциплины

1. Основные типы вычислительных процессов и примеры их алгоритмизации.
 - Линейные и ветвящиеся процессы и их структурное представление
 - Циклические процессы и их структурное представление
2. Базовые структуры данных. Массивы.
 - Одномерные массивы и указатели.
 - Двумерные массивы.
3. Функции.
 - Передача в функцию параметров стандартного типа
 - Передача в функцию массивов, строк, структур.
 - Рекурсивные функции.
4. Строки.
 - Описание строк. Ввод-вывод строк. Функции работы со строками.
5. Структуры.
6. Файлы
 - Описание файлов.
 - Стандартные функции для работы с текстовыми файлами.
 - Стандартные функции для работы с двоичными файлами. Прямой доступ к записям двоичного файла.
7. Динамические структуры данных.
 - Линейные списки.
 - Стеки.
 - Очереди.
 - Бинарные деревья.
8. Классы.
 - Описание классов. Описание объектов.
 - Конструкторы. Конструктор копирования.
 - Перегрузка операций.
9. Наследование.
 - Замещение функций базового класса.
 - Доступ к объектам иерархии.
 - Виртуальные методы.

- Отношения между классами. Диаграммы классов на языке UML.
10. Обработка исключительных ситуаций.
 11. Стандартные потоки.
 - Классы стандартных потоков.
 - Объекты и методы стандартных потоков.
 12. Файловые и строковые потоки.
 13. Стандартная библиотека шаблонов STL.
 - Последовательные контейнеры.
 - Итераторы.
 14. Конечные автоматы.
 - Способы задания конечных автоматов: таблицы и диаграммы.
 - Конечные распознаватели.
 - Пример: константы языка MINI-BASIC.
 15. Лексический блок для языка MINI-BASIC.
 - Множество лексем.
 - Транслитератор.
 - Лексический блок.
 16. Автоматы с магазинной памятью.
 17. Контекстно-свободные и транслирующие грамматики.
 18. Синтаксический блок для языка MINI-BASIC.
 - Множество атомов и транслирующая грамматика.
 - L-атрибутная грамматика.
 - Синтаксический блок.
 19. Генератор кода для MINI-BASIC-компилятора.

III. Образовательные технологии

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Основные типы вычислительных процессов и примеры их алгоритмизации.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Базовые структуры данных. Массивы.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Функции.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Строки.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Структуры.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Файлы	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Динамические структуры данных.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Классы.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Наследование.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Обработка исключительных ситуаций.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Стандартные потоки.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Файловые и строковые потоки.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Стандартная библиотека шаблонов STL.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Конечные автоматы.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Лексический блок для языка MINI-BASIC.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Автоматы с магазинной памятью.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Контекстно-свободные и транслирующие грамматики.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Синтаксический блок для языка MINI-BASIC.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Генератор кода для MINI-BASIC-компилятора.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

Типовые контрольные задания и/или критерии для проверки индикатора ОПК-2.1

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь реализовывать итерационные алгоритмы	<ul style="list-style-type: none"> Дано натуральное число n. Найти сумму нечётных цифр числа или выдать сообщение о том, что таких цифр нет. Дано натуральное число n. Определить является ли это число палиндромом (одинаково читается справа налево и слева направо) Дана последовательность из n целых чисел. Найти минимальный чётный элемент последовательности и порядковый номер этого элемента. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%
Уметь разрабатывать алгоритмы для работы с массивами	<ul style="list-style-type: none"> Дан массив из n целых чисел. Определить идут ли числа в нем в возрастающем порядке. Подсчитать количество столбцов заданной целочисленной матрицы размером $n \times n$, которые составлены из попарно различных элементов. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь пользоваться алгоритмами сортировки и поиска	<ul style="list-style-type: none"> • Дан целочисленный массив, состоящий из n элементов. Перенести в начало массива все чётные положительные элементы, а все остальные в конец, сохраняя исходное взаимное расположение как среди чётных положительных элементов, так и среди остальных элементов (не использовать дополнительный массив). • По заданной квадратной матрице размером $n \times n$ построить вектор длиной $2n - 1$, элементы которого - максимумы элементов диагоналей, параллельных главной диагонали. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%

Типовые контрольные задания и/или критерии для проверки индикатора ОПК-2.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь работать с динамическими структурами	<ul style="list-style-type: none"> • Разработать исполнительную систему «Линейный односвязный список», которая включает следующие операции над списком и его элементами: <ul style="list-style-type: none"> – инициализация списка (сделать список пустым); – проверка списка на пустоту; – добавление нового элемента в начало списка; – добавление нового элемента в конец списка; – поиск в списке элемента, заданного своим значением; – удаление из списка элемента, заданного своим значением; – добавление нового элемента в упорядоченный по возрастанию список; – печать списка; – создание списка. <p>Запрос на выполнение тех или иных действий поступает в диалоговом режиме через меню.</p> <ul style="list-style-type: none"> • Дана непустая последовательность слов, разделённых одним и более пробелом, за последним словом стоит точка. Длина слова не превышает m символов. Описать функцию, которая организует слова в список L. Затем слова, начинающиеся с цифры, направить в упорядоченный по убыванию список L_1, а из списка L эти слова удалить. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%

Типовые контрольные задания и/или критерии для проверки индикатора ОПК-2.3

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь использовать концепцию объектно-ориентированного программирования	<ul style="list-style-type: none"> Составить описание класса для определения одномерных массивов (векторов). Предусмотреть возможность обращения к отдельному элементу массива с контролем выхода за пределы массива, возможность задания произвольных границ индексов при создании объекта, возможность выполнения операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов, вывода на экран элемента массива по заданному индексу, вывода на экран всего массива. Написать программу, демонстрирующую работу с этим классом. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%

Типовые контрольные задания и/или критерии для проверки индикатора ОПК-3.1

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь использовать стандартную библиотеку шаблонов STL.	<ul style="list-style-type: none"> Написать программу для моделирования T-образного сортировочного узла на железной дороге с использованием контейнерного класса <code>stack</code> из STL. Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа). Предусмотреть возможность ввода исходных данных с клавиатуры и из файла. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%

Типовые контрольные задания и/или критерии для проверки индикаторов ОПК-3.2, ОПК-3.3

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь использовать конечные автоматы для обработки текстовой информации	<ul style="list-style-type: none"> Построить конечные распознаватели для описанных ниже множеств цепочек из нулей и единиц. <ol style="list-style-type: none"> Число единиц чётное, а число нулей нечётное Между вхождением единиц чётное число нулей. За каждым вхождением 11 следует 0. Каждый третий символ — единица. 	зачёт — разработан алгоритм и выполнена его программная реализация на 70%
Владеть средствами разработки на языках C/C++	<ul style="list-style-type: none"> Написать shell-сценарий для последовательной трансляции программы с языка C на язык ассемблера, затем — в объектный файл, в конце — скомпоновать его с необходимыми библиотеками Создать статически подключаемую библиотеку функций для сортировки массива целых чисел различными методами Создать динамически подключаемую библиотеку функций для сортировки массива целых чисел различными методами 	зачёт — проделаны основные этапы необходимые для выполнения задания

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Владеть навыками модульной разработки ПО и командной работы	<ul style="list-style-type: none"> • Коллективно разработать и интегрировать модули лексического и синтаксического анализатора • Коллективно разработать программу, имитирующую работу функционального калькулятора, который позволяет выбрать с помощью меню какую-либо из известных ему функций, затем предлагает ввести значение аргумента и, возможно, параметров и после ввода выдаёт соответствующее значение функции. «База знаний» содержит как минимум функции: <ul style="list-style-type: none"> – экспоненту $y = e^x$; – логарифм $y = \ln x$; – интегральный логарифм $y = \text{Li } x$; – ζ-функция $y = \zeta(x)$; – синус $y = \sin x$; – возведение в степень $y = x^a$; – линейную функцию $y = ax + b$. <p>В программе описать базовый класс Function и классы наследники для соответствующих функций</p>	зачёт — каждый из участников выполнил свою часть проекта не менее чем на 50%, выполнена интеграция модулей

V. Учебно-методическое и информационное обеспечение дисциплины

1. Рекомендованная литература

а) Основная литература

- [1] Немцова Т. И. Программирование на языке высокого уровня. Программи- мир. на языке C++: Уч. пос. / Т.И.Немцова и др.; Под ред. Л.Г.Гагариной — М.: ИД ФОРУМ: ИНФРА-М, 2012. — 512 с.: ил.; 60x90 1/16 + CD-ROM. — (Проф. обр.). (п) ISBN 978-5-8199-0492-3 — Режим доступа: <http://www.znanium.com/bookread.php?book=244875> (ЭБС znanium.com)
- [2] Вирт Н. Построение компиляторов [Электронный ресурс] / Никлаус Вирт; пер. с англ. Е. В. Борисов, Л. Н. Чернышов. — М.: ДМК Пресс, 2010. — 192 с.: ил. — Режим доступа: <http://www.znanium.com/bookread.php?book=408433> (ЭБС ИНФРА-М)
- [3] Окулов, С.М. Динамическое программирование [Электронный ресурс] : учеб. пособие / С.М. Окулов, О.А. Пестов. — Электрон. дан. — Москва : Издательство «Лаборатория знаний», 2015. — 299 с. — Режим доступа: <https://e.lanbook.com/book/66114>. — Загл. с экрана.

б) Дополнительная литература

- [4] Вирт, Н. Алгоритмы и структуры данных. Новая версия для Оберона + CD [Электронный ресурс] / Никлаус Вирт; пер. с англ. Ф. В. Ткачев. — М.: ДМК Пресс, 2010. — 272 с.: ил. — Режим доступа:

<http://www.znanium.com/bookread.php?book=408420> (ЭБС znanium.com)

- [5] Залогова, Л.А. Разработка Паскаль-компилятора / Л.А. Залогова. - 4-е изд. (эл.). - Москва : Лаборатория знаний, 2017. - 186 с. - Библиогр.: с. 167. - ISBN 978-5-00101-450-8 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=461267> (27.03.2018).

2. Программное обеспечение

Наименование помещений	Программное обеспечение
Ауд. 4б (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Перечень программного обеспечения (со свободными лицензиями): Adobe Acrobat Reader DC, Anaconda3 2019.07 (Python 3.7.3 64-bit), Apache Tomcat 8.0.27, Cadence SPB/OrCAD 16.6, GlassFish Server Open Source Edition 4.1.1, Google Chrome, IntelliJ IDEA, IIS 10.0 Express, Java SE Development Kit 8 Update 191 (64-bit), JetBrains PyCharm Community Edition 2019.2.1, Kaspersky Endpoint Security для Windows, Lazarus 2.0.12, MiKTeX, NetBeans IDE 8.2, Notepad++ (64-bit x64), ONLYOFFICE Desktop Editors 7.1 (x64), Origin 8.1 Sr2, Python 3.10.7, R for Windows 3.6.1, RStudio Desktop, Visual Studio Community 2022, VLC media player, WinDjView 2.1, Unreal Commander v3.57x64
Ауд. 249 (компьютерный класс №3) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Перечень программного обеспечения (со свободными лицензиями): Adobe Acrobat Reader DC, Google Chrome, Kaspersky Endpoint Security для Windows, ONLYOFFICE Desktop Editors 7.1 (x64), Python 3.10.7, R for Windows 3.6.1, RStudio Desktop, Visual Studio Community 2022, VLC media player, Unreal Commander v3.57x64
Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Перечень программного обеспечения (со свободными лицензиями): Linux Kubuntu, KDE, TeXLive, TeXStudio, LibreOffice, GIMP, Gwenview, ImageMagick, Okular, Skanlite, Google Chrome, KDE Connect, Konversation, KRDC, KTorrent, Thunderbird, Elisa, VLC media player, PulseAudio, KAppTemplate, KDevelop, pgAdmin4, PostgreSQL, Qt, QtCreator, R, RStudio, Visual Studio Code, Perl, Python, Ruby, clang, clang++, gcc, g++, nasm, flex, bison, Maxima, Octave, Dolphin, HTop, Konsole, KSystemLog, Xterm, Ark, Kate, KCalc, Krusader, Spectacle, Vim.

3. Современные профессиональные базы данных и информационные справочные системы

- [1] ЭБС «ZNANIUM.COM» <http://www.znanium.com>
[2] ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru>
[3] ЭБС IPRbooks <http://www.iprbookshop.ru>
[4] ЭБС «Лань» <http://e.lanbook.com>
[5] ЭБС «Юрайт» <https://urait.ru>

- [6] ЭБС ТвГУ <http://megapro.tversu.ru/megapro/Web>
- [7] Научная электронная библиотека eLIBRARY.RU (подписка на журналы)
https://elibrary.ru/projects/subscription/rus_titles_open.asp
- [8] Репозиторий ТвГУ <http://eprints.tversu.ru>

4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- [1] C++ Tutorial, <https://code-live.ru/tag/cpp-manual/>

VI. Методические материалы для обучающихся по освоению дисциплины

Примеры задач для подготовки к контрольным работам

1. Найти все натуральные числа, не превосходящие заданного N , десятичная запись которых есть строго возрастающая или строго убывающая последовательность цифр.
2. Перечислить все натуральные числа, не превосходящие заданного N , в двоичном представлении которых номера ненулевых разрядов образуют арифметическую прогрессию.
3. Найти и распечатать строку заданной целочисленной матрицы размером $n \times n$, в которой длина максимальной серии минимальна.
4. Соседями элемента $a[i][j]$ в матрице назовём элементы $a[k][l]$ с $i - 1 \leq k \leq i + 1$, $j - 1 \leq l \leq j + 1$, $(k, l) \neq (i, j)$. Операция сглаживания матрицы даёт новую матрицу того же размера, каждый элемент которой получается как среднее арифметическое имеющихся соседей соответствующего элемента исходной матрицы. Построить результат сглаживания заданной вещественной матрицы размером $n \times n$.
5. Создать и распечатать текстовый файл. В строке слова разделены более чем одним пробелом. Под словом понимается любая последовательность символов не содержащая пробелов внутри себя. Файл должен содержать не менее 10 строк. Переписать содержимое исходного файла в другой текстовый файл, предварительно преобразовав строки согласно заданному правилу. Распечатать исходный и новый файлы.
 - Удалить из каждой строки слова палиндромы. Если в строке нет таких слов, то записать во второй файл пустую строку.
 - Объединить в отдельную строку через пробел слова нечётной длины, отсортировав их по убыванию длины.
6. Даны целые числа a_1, a_2, \dots, a_n , целочисленная квадратная матрица b . Заменить нулями в матрице b те элементы с четной суммой индексов ($i + j - \text{чётное}$),

для которых имеются равные среди a_1, a_2, \dots, a_n . Исходную матрицу оставить без изменения. Рассматривая строки матрицы как систему векторов, определить является ли система векторов ортогональной для матрицы, полученной в результате преобразования исходной. В программе использовать:

- функцию преобразования матрицы;
 - функцию, определяющую есть ли в целочисленном векторе элемент равный заданному числу;
 - функцию, определяющую ортогональность.
7. Описать класс для работы с кольцевым списком. Написать программу, демонстрирующую работу с этим классом. «Считалка». N ребят располагаются по кругу. Начав отсчёт от первого, удаляют каждого K -го, смыкая круг после каждого удаления. Определить, кто остался и порядок удаления ребят из круга.
 8. Построить конечный автомат для констант языка MINI-BASIC и выполнить его программную реализацию.

Требования к рейтинг контролю (1 семестр)

Контрольная работа 1. Тема: массивы в языке C/C++. Пример задания:

1. Дан целочисленный массив x из n элементов. Вычислить сумму

$$s = x[1] * x[n] + x[2] * x[n - 1] + \dots + x[n - 1] * x[2] + x[n] * x[1].$$

2. Даны два массива a и b размера n . Найти наименьший из тех элементов массива a , которые не входят в массив b .
3. Дан целочисленный массив a из n элементов. Осуществить циклический сдвиг элементов массива на k позиций вправо. Не использовать дополнительный массив.
4. Дан целочисленный массив из n элементов. Поменять порядок следования элементов, находящихся между минимальным и максимальным элементами на обратный. Предполагается, что минимальный и максимальный элементы в массиве единственные.
5. Дана целочисленная матрица размерности $n \times m$. Подсчитать количество строк в матрице, элементы которых упорядочены по убыванию. Если таких строк нет, то выдать сообщение. За решение каждой задачи выставляется максимум 10 баллов.

Контрольная работа 2. Тема: функции в языке C/C++. Пример задания:

1. Написать функцию, которая для заданного числа n определяет, является ли это число полным квадратом.
2. Заданный массив целых чисел делится на три части двумя элементами: максимальным и минимальным, Определите сумму элементов в каждой части

массива. Использовать функции для нахождения индексов минимального и максимального элементов и подсчёта суммы элементов в указанной части массива.

3. Дано натуральное число m , последовательность целых чисел a_1, a_2, \dots, a_m и целочисленная квадратная матрица порядка m . Строку с номером i назовём отмеченной, если $a_i > 0$, и неотмеченной, в противном случае. Построить вектор v из первых положительных элементов отмеченных строк. Если отмеченная строка не содержит положительных элементов, то положить в вектор 0. Если матрица не содержит отмеченных строк, то выдать сообщение. В программе описать:
 - функцию, которая выделяет память под матрицу и заполняет ее с ввода;
 - функцию, которая находит первый положительный элемент в векторе;
 - функцию, которая строит по матрице вектор.
4. Дана целочисленная матрица размерности $n \times m$. Подсчитать количество строк в матрице, которые упорядочены (как по возрастанию так и по убыванию). В программе использовать аппарат функций.
5. Дана целочисленная матрица размерности $n \times m$. Подсчитать количество строк в матрице, элементы которых образуют арифметическую прогрессию. В программе использовать аппарат функций. За решение каждой задачи выставляется максимум 10 баллов.

Требования к рейтинг контролю (2 семестр)

Контрольная работа 1. Темы: строки, текстовые файлы. Пример задания:

1. Написать функцию с именем `strins`, которая вставляет в строку s_1 с k -ой позиции строку s_2 .
2. Написать функцию, которая записывает из строки s_1 в строку s_2 через пробел слова чётной длины, в которых нет ни одной повторяющейся буквы. Слова в строке s_1 представляют собой последовательность латинских букв и разделены одним или несколькими пробелами.
3. Дан текст, состоящий не более, чем из 300 символов, в котором нет вхождения символов «(» и «)». Выполнить его сжатие, то есть заменить всякую максимальную подпоследовательность, составленную из более чем трёх вхождений одного и того же символа, на $(k)s$, где s — повторяемый символ, а $k > 3$ — количество его повторений.
4. Описать структуру с именем `Знак`, содержащую следующие поля:
 - фамилия;
 - имя;
 - знак Зодиака;
 - дата рождения (массив из трёх чисел).

Написать функцию, которая читает данные из текстового файла и записывает в выходной текстовый файл информацию о людях, родившихся под знаком, название которого вводится с клавиатуры.

5. В текстовом файле записана матрица. Первая строка файла содержит размер (в виде $n * m$), следующие n строк содержат построчно элементы матрицы. Написать программу, располагающую строки матрицы по возрастанию их характеристик (сумма чётных положительных элементов). В программе описать следующие функции:

- Вычисление характеристики для одномерного массива (строка)
- Построение вектора характеристик
- Сортировка методом вставок
- Чтение матрицы из файла
- Вывод матрицы в текстовый файл

За решение каждой задачи выставляется максимум 10 баллов.

Контрольная работа 2. Тема: динамические структуры данных в языке C/C++. Пример задания:

1. Разработать исполнительную систему «Линейный односвязный список», которая включает следующие операции над списком и его элементами: инициализация списка (сделать список пустым);

- проверка списка на пустоту;
- добавление нового элемента в начало списка;
- добавление нового элемента в конец списка;
- поиск в списке элемента, заданного своим значением;
- удаление элемента, заданного своим значением;
- добавление нового элемента в упорядоченный по возрастанию список;
- печать списка;
- создание списка.

Запрос на выполнение тех или иных действий поступает в диалоговом режиме через меню.

2. Описать функцию, которая переворачивает список, т. е. изменяет ссылки в этом списке так, чтобы элементы оказались расположенными в обратном порядке. (другой список не создавать)

3. Написать функцию, которая в конец непустого двухсвязного линейного списка добавляет все его элементы, располагая их в обратном порядке (например, по списку из элементов 1, 5, 3 требуется построить список 1, 5, 3, 3, 5, 1).

4. В текстовом файле записана без ошибок формула следующего вида:
 $\langle \text{формула} \rangle ::= \langle \text{цифра} \rangle \mid M(\langle \text{формула} \rangle, \langle \text{формула} \rangle) \mid m(\langle \text{формула} \rangle, \langle \text{формула} \rangle)$
 $\langle \text{цифра} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$
 где «M» обозначает функцию max, а «m» — min. Вычислить (как целое число) значение данной формулы (например, $M(5, m(6, 8)) \rightarrow 6$). В программе использовать стек. Вычисление значения данной формулы оформить в виде функции.
5. Пусть даны две очереди x и y , содержащие вещественные числа. Из каждой очереди одновременно извлекается по одному числу x и y соответственно. Если $x < y$, то число $(x + y)$ помещается в конец очереди x , иначе число $(x - y)$ помещается в конец очереди y . Необходимо определить число шагов, через которое одна из очередей станет пустой. За решение каждой задачи выставляется максимум 10 баллов.

Требования к рейтинг контролю (3 семестр)

Контрольная работа 1. Темы: классы, перегрузка операций, наследование, обработка исключительных ситуаций, потоки. Пример задания:

1. Составить описание класса для определения одномерных массивов целых чисел. Предусмотреть возможность обращения отдельному элементу массива с контролем выхода за границы массива, возможность выполнения операций поэлементного сложения и вычитания массивов одинаковой размерности, вывод на экран элемента массива по заданному индексу, перегрузку операций ввода и вывода через стандартные потоки, обработку исключений. Написать программу, демонстрирующую работу с этим классом. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.
2. Описать класс «домашняя библиотека». Предусмотреть возможность работы с произвольным числом книг, поиска книги по какому либо признаку (например, по автору или по году издания), добавление книг в библиотеку, удаление книг из нее, сортировки книг по разным полям. Написать программу, демонстрирующую работу с этим классом. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.
3. Описать класс «вещественная матрица». Класс должен реализовывать следующие операции над матрицами:
 - сложение, вычитание, умножение (+, -, *);
 - операции присваивания (=, + =);
 - операции сравнения на равенство и неравенство (==, !=);
 - операции ввода/вывода в стандартные потоки;
 - методы вычисления детерминанта и нормы.

Написать программу, демонстрирующую работу с этим классом. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса. За решение каждой задачи выставляется максимум 20 баллов.

Контрольная работа 2. Темы: стандартная библиотека шаблонов STL. Пример задания:

1. (15 баллов) Написать программу, моделирующую управление каталогом в файловой системе. Для каждого файла в каталоге хранится следующая информация: имя файла, дата создания, количество обращений к файлу. Программа должна обеспечивать:

- начальное формирование каталога файлов;
- вывод каталога файлов;
- удаление файлов, дата создания которых раньше заданной;
- выборку файла с наибольшим количеством обращений.

Выбор моделируемой функции осуществляется с помощью меню. Для представления каталога использовать контейнерный класс `list` из STL.

2. (15 баллов) Написать программу «Англо-русский и русско-английский словарь». «База данных» словаря должна содержать синонимичные варианты перевода слов. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

- загрузка «базы данных» словаря (из файла);
- выбор режима работы:
- англо-русский;
- русско-английский;
- вывод вариантов перевода заданного английского слова;
- вывод вариантов перевода заданного русского слова.

Базу данных словаря реализовать в виде двух контейнеров типа `map`. За решение каждой задачи выставляется максимум 15 баллов.

3. (10 баллов) Написать программу формирования частотного словаря появления отдельных слов в некотором тексте. Исходный текст читается из файла, результат — частотный словарь — записывается в выходной файл. Частотный словарь реализовать в виде контейнера типа `map`.

Требования к рейтинг контролю (4 семестр)

Контрольная работа 1. Тема: конечные автоматы. Пример задания:

1. (10 баллов) Постройте конечный автомат с входным алфавитом $\{0, 1\}$, который допускает тв точности такое множество цепочек:

- входную цепочку 101;
 - две входных цепочки: 01 и 0100;
 - все цепочки, начинающиеся с 0 и кончающиеся на 1;
 - все цепочки, в которых перед и после 1 стоит 0.
2. (5 баллов) Постройте конечный автомат, который будет распознавать следующие зарезервированные слова: STEP, STRING, SWITCH.
 3. (5 баллов) Постройте конечный автомат, который будет распознавать химические формулы, составленные из восьми элементов: *H, C, N, O, SI, S, CL* и *SN*. Элементы в формулах разделяются запятыми. Они могут появляться в любом порядке и в любых сочетаниях. Формулы необязательно представляют реально существующие соединения. Вот несколько образцов формул: *H2, O; O, H7; SN, S, O4; N, H4, C7, H5, O2; O2* Имеется девять входных символов: *C, H, I, L, N, O, S*, а также — цифры.

Практическое задание по построению компилятора для языка MINI-BASIC Темы: лексический блок, синтаксический блок. Пример задания:

1. Выполнить программную реализацию конечного автомата для лексического блока, решающего следующие проблемы идентификации: обнаружение зарезервированных слов, идентификация переменных, идентификация номеров строк, идентификация знаков отношений.
2. Выполнить программную реализацию конечного автомата для синтаксического блока, построенного на основе L-атрибутивной транслирующей грамматики с входной LL(1)-грамматикой. За решение каждой задачи выставляется максимум 40 баллов.

VII. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для аудиторной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 308 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран проектор.

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 205 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран, проектор.
Ауд. 318 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран, проектор.
Ауд. 310 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели.
Ауд. 46 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Компьютер, экран, проектор, кондиционер.
Ауд. 249 (компьютерный класс №3) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, компьютеры, проектор.

Для самостоятельной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, доска маркерная, компьютер, сервер (системный блок), концентратор сетевой.

VIII. Сведения об обновлении рабочей программы дисциплины

№ п/п	Обновленный раздел рабочей программы дисциплины	Описание внесённых изменений	Дата и протокол заседания кафедры, утвердившего изменения
1	I. Аннотация. IV. Оценочные материалы для проведения текущей и промежуточной аттестации	Изменения в учебные планы и в рабочие программы дисциплин, формирующих новые/измененные компетенции в соответствии с приказом Минобрнауки России от 26.11.2020 г. № 1456	Протокол № 7 заседания ученого совета от 30.12.2021 года
2	V. Учебно-методическое и информационное обеспечение дисциплины	Внесены изменения в программное обеспечение	От 29.09.2022 года, протокол № 2 ученого совета факультета
3	VII. Материально-техническое обеспечение	Внесены изменения в материально-техническое обеспечение аудиторий	От 29.09.2022 года, протокол № 2 ученого совета факультета