

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Смирнов Сергей Николаевич
Должность: врио ректора
Дата подписания: 26.10.2023 15:41:26
Уникальный программный ключ:
69e375c64f7e975d4e8830e7b4fct2ad1bf35f08

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тверской государственной университет»

УТВЕРЖДАЮ
Руководитель ООП

« 30 » октября 2023 г.



Рабочая программа дисциплины (с аннотацией)
Практикум на ЭВМ

Направление подготовки
15.03.06 МЕХАТРОНИКА И РОБОТОТЕХНИКА

Профиль подготовки
Интеллектуальное управление в мехатронных и робототехнических системах

Для студентов 1, 2-го курса
Формы обучения - очная

Составитель(и):

- к.т.н. доц. Волушкова В.Л.

1. Аннотация

1. Цель и задачи дисциплины:

Выработать у обучающихся практические навыки по проектированию алгоритмов и их программной реализации на языках высокого уровня (C/C++).

2. Место дисциплины в структуре ООП

Дисциплина входит в раздел «Информатика и информационно-коммуникационные технологии» обязательной части блока 1.

Предварительные знания и навыки. знание школьных курсов математики и информатики.

Дальнейшее использование. Полученные знания используются в последующем при изучении предметов: «Технология программирования», «Базы данных». Знания, умения и навыки, полученные при изучении дисциплины, закрепляются в конце каждого семестра учебной практикой.

3. Объем дисциплины: зач. ед.– 5; всего 180 акад. часов, в том числе:

контактная аудиторная работа практических занятий 47 часов, в том числе практическая подготовка 47 часов, лабораторных занятий 62 часа, в том числе практическая подготовка 62 часа,

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

самостоятельная работа 71 час., в том числе контроль 0 часов.

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-1, Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-1.1, Способен применять естественнонаучные и общетеоретические знания, методы математического анализа и моделирования в профессиональной деятельности ОПК-1.3, Применяет методы математического и компьютерного моделирования, средства автоматизированного проектирования в теоретических и расчетно-экспериментальных исследованиях

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ОПК-11 Способен разрабатывать и применять алгоритмы и современные цифровые программные методы расчетов и проектирования отдельных устройств и подсистем мехатронных и робототехнических систем с использованием стандартных исполнительных и управляющих устройств, средств автоматики, измерительной и вычислительной техники в соответствии с техническим заданием, разрабатывать цифровые алгоритмы и программы управления робототехнических систем	ОПК-11.2 Разрабатывает программное обеспечение для управления мехатронными и робототехническими системами
ПК-1 Способен участвовать в качестве исполнителя в научно-исследовательских разработках новых робототехнических и мехатронных систем	ПК-1.2 Разрабатывает экспериментальные макеты управляющих, информационных и исполнительных модулей мехатронных и робототехнических систем и проводит их экспериментальное исследование с применением современных информационных технологий
ОПК-14 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-14.1 Применяет алгоритмы и современные цифровые программные методы расчетов и проектирования деталей и модулей мехатронных и робототехнических систем ОПК-14.2 Разрабатывает программное обеспечение для обработки информации в мехатронных и робототехнических системах

5. Форма промежуточной аттестации и семестр прохождения:

зачет в 1–2 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.)				Контроль сам. раб., в т.ч. кур- совая работа	Сам. раб., в т.ч. контроль (час.)
		Лекции		Практ. заня- тия / Лаб. работы			
		Всего	В т.ч. практ подг.	Всего	В т.ч. практ подг.		
1	2	3	4	5	6	7	8
Синтаксис, базовые типы переменных, операции, операторы, стандарт языка C	4	0	0	1/2	1/2	0	1
Системы счисления. Поразрядные операции.	15	0	0	3/6	3/6	0	6
Функции в C	27	0	0	5/10	5/10	0	12
Указатели и ссылки	26	0	0	5/10	5/10	0	11
Массивы и строки в C	20	0	0	7/8	7/8	0	5
Файловый ввод/вывод.	15	0	0	2/2	2/2	0	11
Работа с текстом	20	0	0	8/8	8/8	0	4
Сортировка и поиск	26	0	0	8/8	8/8	0	10
Динамические структуры данных	27	0	0	8/8	8/8	0	11
Итого	180	0	0	47/62	47/62	0	71

Учебная программа дисциплины

1. Синтаксис, базовые типы переменных, операции, операторы, стандарт языка C

- Переменные и константы.
- Функции. Форматный ввод/вывод на консоль.
- Операции. Оператор присваивания.
- Операторы. Условные операторы. Операторы выбора. Операторы цикла.

2. Системы счисления. Поразрядные операции.

- Двоичная, восьмеричная, шестнадцатеричная система счисления.
- Поразрядные операции. Двоичный дополнительный код.
- Преобразование типов при присваивании. Переполнение целых.

3. Функции в C

- Функции, возвращаемые значения, параметры и аргументы.
- Объявление и определение функций. Локальные и глобальные переменные.

- Дополнительные сведения о функциях. Рекурсия. Стек и рекурсия.

4. Указатели и ссылки

- Указатель – средство хранения адреса. Имена указателей. Оператор разыменовывания. Использование адреса, хранящегося в указателе.
- Память стековая и динамически распределяемая. Утечка памяти.
- Вычисления с указателями. Передача аргументов функций как указателей.
- Ссылки в C++

5. Массивы и строки в C

- Одномерные массивы. Инициализация массива. Индексирование.
- Массивы и указатели. Передача одномерного массива в функцию.
- Двоичный поиск в отсортированном массиве.
- Строки. Библиотечные функции для работы со строками.
- Многомерные массивы. Передача многомерного массива в функцию.
- Массивы указателей. Динамические массивы.

6. Файловый ввод/вывод.

- Потоки и файлы. Основы файловой системы.
- Указатель файла, открытие файла, закрытие файла, запись символа, чтение символа и т.д.
- Буферный ввод/вывод. Функции `fread()` и `fwrite()`.
- Ввод/вывод при прямом доступе. Функция `fseek()`.
- Структуры и бинарные файлы. 7. Работа с

текстом

- Методы кодирования символов. Кодировки ASCII, ANSI, OEM, UTF, Unicode
- Указатели, действия с указателями. Работа с массивами при помощи указателей
- Простейшие алгоритмы: определение длины, копирование, конкатенация
- Поиск и замена текста. Работа с различными элементами текста: символами, словами, строками

8. Сортировка и поиск

- Задача хранения информации. Линейный поиск. Максимальные и средние оценки времени для линейного поиска
- Двоичный поиск и его разновидности. Оценки времени для двоичного поиска. Поиск методом Фибоначчи, золотого сечения. Интерполяционный поиск и его обобщения
- Задача сортировки. Сортировка пузырьком. Временные оценки пузырьковой сортировки

- Сортировка при помощи поиска максимального элемента. Временные оценки
- Сортировка вставками. Оценки времени.
- Сортировка почти упорядоченных массивов. Модификации методов.
- Сортировки Шелла. Задача выбора последовательности шагов. Оценки для различных последовательностей
- Слияние упорядоченных массивов. Модификации алгоритма для нахождения пересечения, объединения и разности множеств
- Сортировка слиянием. Временные оценки
- Быстрая сортировка Хоара. Оценки времени

9. Динамические структуры данных

- Операции динамического распределения памяти. Временные оценки операций
- Динамические структуры данных
- Линейные списки. Односвязные и двусвязные списки
- Основные операции со списками: добавление и удаление элемента
- Деревья. Двоичные деревья поиска
- Операции над двоичными деревьями поиска: вставка и удаление элемента, поиск
- Сбалансированные деревья. Методы балансировки
- Хеш-таблицы. Основная концепция, выбор хеш-функции
- Методы разрешения коллизий: метод цепочек, линейного опробования, двойного хеширования

III. Образовательные технологии

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Синтаксис, базовые типы переменных, операции, операторы, стандарт языка C	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Системы счисления. Поразрядные операции.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Функции в С	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Указатели и ссылки	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Массивы и строки в С	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Файловый ввод/вывод.	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Работа с текстом	практические занятия, лабораторные занятия	решение задач, работа с программным обеспечением, разработка программного обеспечения
Сортировка и поиск	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

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

Типовые контрольные задания и/или критерии для проверки индикаторов ОПК-1.1, ОПК-1.3, ОПК-11.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания

<p>Уметь писать программы, используя синтаксис, базовые типы переменных, операции, операторы</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Напишите функцию, которая подсчитывает количество единиц в двоичной записи целого числа, используя битовые операции. • Написать функцию, которая находит элемент линейного массива 	<p>зачтено умеет писать функции, выполнять задания по чтению записи текстового файла.</p>
<p>Требования к обучающемуся</p>	<p>Типовые контрольные задания для оценки знаний, умений, навыков</p>	<p>Показатели и критерии оценивания, шкала оценивания</p>
<p>соответствующий стандарту языка C</p>	<p>сыва, наиболее приближенный к среднему арифметическому элементов массива.</p> <ul style="list-style-type: none"> • Написать функцию, которая вычисляет в одномерном массиве, состоящем из n вещественных элементов, сумму элементов массива, расположенных между первым и последним нулевыми элементами. • Написать на интервале $[x, x + \text{количество строк}]$ программу, которая находит количество простых чисел. • Написать функцию, получающую в качестве аргумента имя файла, содержащего последовательность вещественных чисел неизвестной длины, и возвращающую целое число, равное количеству максимальных элементов этой последовательности. Функция должна возвращать -1, -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д.. Основная программа должна вызывать эту функцию и выводить на экран результат ее работы. 	
<p>Уметь создавать массивы и строки.</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Написать функцию, которая находит элемент линейного массива, наиболее приближенный к среднему арифметическому элементов массива. • Написать функцию, которая вычисляет в одномерном массиве, состоящем из n вещественных элементов, сумму элементов массива, расположенных между первым и последним нулевыми элементами. • Написать функцию, которая переворачивает строку. Например: "главрыба" — "абырвалг". • Функция «обрезает» строку на первом символе, 	<p>зачтено — умеет создавать массивы и строки, может передать массив и строку в функцию.</p>

	совпадающем с данным символом.	
Уметь самостоятельно использовать библиотечные функции для работы со строками.	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Написать программу, которая считывает текст из файла и выводит на экран только строки, содержащие двузначные числа. • Написать программу, которая считывает английский текст из файла и выводит на экран слова, начинающиеся с гласных букв. • Написать программу, которая считывает текст из файла и выводит его на экран, меняя местами каждые два соседних слова. 	зачтено—умеет пользоваться основными функциями из библиотеки string.h
Уметь работать с текстом, знать методы кодирования символов, уметь работать с кодировками ASCII, ANSI, OEM, UTF, Unicode	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Написать функцию <code>void f(char * str)</code>, которая в строке <code>str</code> удаляет все кратные пробелы между словами, а также в начале и конце строки. Например, из строки « Quando tu sei vicina » должно получиться «Quando tu sei vicina». Вычислить максимальное время работы функции в зависимости от длины входных данных. • Написать функцию, которая считывает текст из файла и выводит на экран только предложения, не содержащие запятых. 	зачтено — умеет работать с текстом в кодировке ASCII, может работать с кодировкой Unicod.
Знать и уметь применять алгоритмы сортировки и	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> • Написать программу, сортирующую методом вставок главную диагональ двумерного массива целых чисел. Массив считать 	зачтено — умеет применять алгоритмы сортировки и поиска.

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
<p>поиска, а именно сортировки вставками, сортировки пузырьком, сортировки Шелла. Знать основные алгоритмы поиска в отсортированно массиве</p>	<p>из текстового файла, содержащего в первой строке кол-во строк и кол-во столбцов, а далее в каждой строке целые числа, соответствующие элементам строки массива.</p> <ul style="list-style-type: none"> • Написать программу, сортирующую методом вставок двумерный массив целых чисел. При этом самое маленькое число должно оказаться в первом элементе первой строки, а самое большое в последнем элементе последней строки. • Написать программу, ищущую методом половинного деления строку с заданным значением 1-го элемента строки в двумерном массиве целых чисел. Выполнить все необходимые для этого операции. Не использовать стандартные функции. 	
<p>Уметь применять алгоритмы сортировка пузырьком, вставками, сортировку Шелла, созданные в командной разработке, для упорядочения массивов и списков.</p>	<p>Примеры задач для самостоятельных работ:</p> <ul style="list-style-type: none"> • Отсортировать методом вставок список, содержащий целые числа. • Отсортировать сортировкой Шелла, созданной в группе, список, содержащий целые числа. <p>Примеры задач для самостоятельных работ (задание для групповой работы):</p> <p>Создать и протестировать контейнер для хранения данных о студенческой группе. Предусмотреть возможность быстрого поиска информации о студенте группы :</p> <ul style="list-style-type: none"> • Структурировать информацию о группе и о студентах группы (выполняет руководитель группы) • Выбрать способ хранения данных (выполняет руководитель группы) • Выбрать способ написания сортировки (с функциями сравнения или без)(выполняет руководитель группы) • Распределить среди участников группы задания, которые необходимо выполнить для решения поставленной задачи (выполняет 	<p>зачтено — умеет применять сортировки для упорядочения массива целых чисел, умеет применять шаблонные сортировки для упорядочивания массива, умеет создавать программы в группе разработки.</p>

	<p>руководитель группы)</p> <ul style="list-style-type: none"> • Создать программу для хранения информации о студентах (выполняет участник группы) • Создать программу, которая заполняет контейнер данными из текстового файла (выполняет участник группы) • Создать функции, реализующие сортировки пузырьком, вставками, сортировку Шелла (выполняют участники группы) • Создать тесты, которые проверяют правильность работы программы (выполняет участник группы) • Выполнить сравнение эффективности алгоритмов сортировки (выполняет участник группы) • Создать функции, реализующие поиск информации в массиве данных о студентах (по фамилии студента) • Создать функции, реализующие поиск информации в массиве данных о студентах (по фамилии дате рождения студента) • Выполнить тестирование контейнера для хранения данных о студенческой группе (выполняют все участники группы) 	
--	---	--

Типовые контрольные задания и/или критерии для проверки индикаторов ОПК-14.1, ОПК-14.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
<p>Уметь применять алгоритмы сортировка пузырьком, вставками, сортировку Шелла, созданные в командной разработке, для</p>	<p>Примеры задач для самостоятельных работ:</p> <ul style="list-style-type: none"> • Отсортировать методом вставок список, содержащий целые числа. • Отсортировать сортировкой Шелла, созданной в группе, список, содержащий целые числа. <p>Примеры задач для самостоятельных работ (задание для групповой работы):</p> <p>Создать и протестировать контейнер для хранения данных о студенческой группе.</p> <p>Предусмотреть возможность быстрого поиска</p>	<p>зачтено — умеет применять сортировки для упорядочения массива целых чисел, умеет применять шаблонные сортировки для упорядочивания массива, умеет</p>

<p>упорядочения массивов и списков.</p>	<p>информации о студенте группы :</p> <ul style="list-style-type: none"> • Структурировать информацию о группе и о студентах группы (выполняет руководитель группы) • Выбрать способ хранения данных (выполняет руководитель группы) • Выбрать способ написания сортировки (с функциями сравнения или без) • (выполняет руководитель группы) • Распределить среди участников группы задания, которые необходимо выполнить для решения поставленной задачи (выполняет руководитель группы) • Создать программу для хранения информации о студентах (выполняет участник группы) • Создать программу, которая заполняет контейнер данными из текстового файла (выполняет участник группы) • Создать функции, реализующие сортировки пузырьком, вставками, сортировку Шелла (выполняют участники группы) • Создать тесты, которые проверяют правильность работы программы (выполняет участник группы) • Выполнить сравнение эффективности алгоритмов сортировки (выполняет участник группы) • Создать функции, реализующие поиск информации в массиве данных о студентах (по фамилии студента) • Создать функции, реализующие поиск информации в массиве данных о студентах (по фамилии дате рождения студента) • Выполнить тестирование контейнера для хранения данных о студенческой группе (выполняют все участники группы) 	<p>создавать программы в группе разработки.</p>
---	---	---

<p>Знать динамические структуры данных, а именно линейные списки, односвязные и двусвязные списки. Уметь использовать динамические данные, созданные в группе</p>	<p>Примеры вопросов к зачету:</p> <ul style="list-style-type: none"> • Линейные списки. Основные операции со списками: добавление и удаление элемента • Двусвязные списки. Основные операции со списками: добавление и удаление элемента • Сцепление и расцепление списков. • Кольцевые списки 	<p>зачтено—знает что такое динамические структуры данных, может создать линейный список, умеет создавать программы в группе разработки.</p>
<p>Требования к обучающемуся</p>	<p>Типовые контрольные задания для оценки знаний, умений, навыков</p>	<p>Показатели и критерии оценивания, шкала оценивания</p>
	<p>• Двумерные списки. Представление матриц двумерными списками.</p> <p>Примеры задач для самостоятельных работ (задание для групповой работы): Создать и протестировать контейнер для хранения данных о студенческой группе:</p> <ul style="list-style-type: none"> • Структурировать информацию о группе и о студентах группы (выполняет руководитель группы) • Выбрать способ хранения данных (выполняет руководитель группы) • Распределить среди участников группы задания, которые необходимо выполнить для решения поставленной задачи (выполняет руководитель группы) • Создать шаблонный список, структуру которого определил руководитель (выполняет участник группы) • Создать программу, которая заполняет контейнер данными из текстового файла (выполняет участник группы) • Создать тесты, которые проверяют правильность работы программы (выполняет участник группы) 	

	<ul style="list-style-type: none"> • Выполнить тестирование контейнера для хранения данных о студенческой группе (выполняют все участники группы) 	
Знать как создать двоичное дерево поиска	<p>Примеры вопросов к зачету:</p> <ul style="list-style-type: none"> • Операции над двоичными деревьями поиска: вставка и удаление элемента, поиск • Сбалансированные деревья. Методы балансировки 	оценка 3 — умеет создать двоичное дерево поиска, оценка 4 — умеет выполнять операции над двоичными деревьями поиска, оценка 5 — умеет создавать сбалансированные деревья поиска.
Знать понятие хеш-таблица, основную концепцию, как вы- брать хеш-функцию	<p>Примеры вопросов к зачету:</p> <ul style="list-style-type: none"> • Хеш-таблицы. Основная концепция, выбор хеш-функции • Методы разрешения коллизий: метод цепочек, линейного опробования, двойного хеширования. 	зачтено — может осуществить поиск данных в массиве с помощью хеш-таблицы.

Типовые контрольные задания и/или критерии для проверки индикатора ПК-1.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь разрабатывать макеты Хеш-таблицы для поиска информации в исполнительных модулях	<p>Примеры задач для самостоятельных работ:</p> <ul style="list-style-type: none"> • Создать хеш-таблицу для поиска информации, хранящейся в исполнительном модуле. • Осуществить поиск информации, необходимый исполнительному модулю. 	зачтено — может осуществить поиск информации с использованием хеш-таблиц.

<p>Уметь разрабатывать макеты двоичного дерева поиска для поиска информации в исполнительных модулях</p>	<p>Примеры задач для самостоятельных работ:</p> <ul style="list-style-type: none"> • Создать двоичное дерево поиска для информации, хранящейся в исполнительном модуле. • Протестировать алгоритм создания двоичного дерева поиска целых чисел. Оценить время поиска элемента, время вставки элемента. 	<p>зачтено — может осуществлять поиск информации с помощью двоичного дерева поиска.</p>
<p>Уметь тестировать двоичное дерево поиска по заданному сценарию</p>	<p>Примеры задач для самостоятельных работ:</p> <ul style="list-style-type: none"> • Протестировать алгоритм создания двоичного дерева поиска целых чисел. Применить двоичное дерево для сортировки набора целых чисел. Оценить время сортировки в сравнении с сортировкой вставками. Сформировать сценарий тестирования. • Протестировать алгоритм создания двоичного дерева поиска целых чисел. Оценить время поиска элемента в сравнении с алгоритмом поиска в отсортированном массиве. Сформировать сценарий тестирования. 	<p>зачтено — умеет сформировать сценарий тестирования..</p>

V. Учебно-методическое и информационное обеспечение дисциплины

1. Рекомендованная литература

а) Основная литература

- [1] Немцова Т. И. Программирование на языке высокого уровня. Програм- 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. Черны- шов. — 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.znaniium.com/bookread.php?book=408420> (ЭБС znaniium.com)

2. Программное обеспечение

Компьютерный класс факультета прикладной математики и кибернетики № 46 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	
Adobe Acrobat Reader DC - Russian	бесплатно
Apache Tomcat 8.0.27	бесплатно
Cadence SPB/OrCAD 16.6	Государственный контракт на поставку лицензионных программных продуктов 103 - ГК/09 от 15.06.2009
GlassFish Server Open Source Edition 4.1.1	бесплатно
Google Chrome	бесплатно
Java SE Development Kit 8 Update 45 (64-bit)	бесплатно
JetBrains PyCharm Community Edition 4.5.3	бесплатно
JetBrains PyCharm Edu 3.0	бесплатно
Kaspersky Endpoint Security 10 для Windows	Акт на передачу прав ПК545 от 16.12.2022
Lazarus 1.4.0	бесплатно
Mathcad 15 Mo10	Акт предоставления прав ИС00000027 от 16.09.2011
MATLAB R2012b	Акт предоставления прав № Ус000311 от 25.09.2012
Многофункциональный редактор ONLYOFFICE бесплатное ПО	бесплатно
ОС Linux Ubuntu бесплатное ПО	бесплатно
MiKTeX 2.9	бесплатно
MSXML 4.0 SP2 Parser and SDK	бесплатно
NetBeans IDE 8.0.2	бесплатно
NetBeans IDE 8.2	бесплатно
Notepad++	бесплатно
Oracle VM VirtualBox 5.0.2	бесплатно
Origin 8.1 Sr2	договор №13918/М41 от 24.09.2009 с ЗАО «СофтЛайн Трейд»
Python 3.1 pygame-1.9.1	бесплатно
Python 3.4 numpy-1.9.2	бесплатно
Python 3.4.3	бесплатно
Python 3.5.1 (Anaconda3 2.5.0 64-bit)	бесплатно
WCF RIA Services V1.0 SP2	бесплатно
WinDjView 2.1	бесплатно
R Studio	бесплатно
Anaconda3 2019.07 (Python 3.7.3 64-bit)	бесплатно

<p>Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)</p>	<p>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</p>
---	--

3) Современные профессиональные базы данных и информационные справочные системы

1. ЭБС «ZnaniUM.COM» www.znanium.com;
2. ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru/>;
3. ЭБС «Лань» <http://e.lanbook.com>.

4) Перечень ресурсов информационно-телекоммуникационной сети «Ин-тернет», необходимых для освоения дисциплины
Интернет-университет <http://www.intuit.ru>

VI. Методические материалы для обучающихся по освоению дисциплины

Примеры задач для подготовки к контрольным работам

1. Примеры задач для контрольных работ:

- Написать функцию `void f(char * str)`, которая в строке `str` убирает все кратные пробелы между словами, а также в начале и конце строки. Например, из строки «`Quando tu sei vicina`» должно получиться «`Quando tu sei vicina`». Вычислить максимальное время работы функции в зависимости от длины входных данных.
- Написать шаблонную функцию `template<class T> int f(T * a, int m, T * b, int n)` которая принимает на вход массивы `a` и `b` длиной `m` и `n` соответственно. Функция должна вернуть количество элементов, которые встречаются только в этих массивах. Максимальное время работы функции должно быть $O(n \log^2 m)$. В одном

Требования к рейтинг контролю (1 семестр)

Самостоятельная работа 1. Темы: программы, использующие синтаксис, базовые типы переменных, операции, операторы соответствующие стандарту языка C. Пример задания:

1. Написать функцию, которая находит элемент линейного массива, наиболее приближенный к среднему арифметическому элементов массива.
2. Написать функцию, которая выводит на печать шестнадцатеричное представление десятичного числа, используя двоичные операции
3. Написать программу, которая находит количество простых чисел на интервале $[x, y)$.

За решение каждой задачи выставляется максимум 5 баллов.

Самостоятельная работа 2. Темы: Библиотечные функции для работы со строками. Пример задания:

1. Написать программу, которая считывает текст из файла и выводит на экран только строки, содержащие двузначные числа.
2. Написать программу, которая считывает английский текст из файла и выводит на экран слова, начинающиеся с гласных букв.
3. Написать функцию `void f(char * str)`, которая в строке `str` убирает все кратные пробелы между словами, а также в начале и конце строки. Например, из строки «`Quando tu sei vicina`» должно получиться «`Quando tu sei vicina`».

За решение каждой задачи выставляется максимум 5 баллов.

Самостоятельная работа 3. Темы: файловый ввод/вывод. Пример задания:

1. Написать функцию, получающую в качестве аргумента имя файла, содержащего последовательность вещественных чисел неизвестной длины, и возвращающую целое число, равное количеству максимальных элементов этой последовательности. Функция должна возвращать -1, -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д.. Основная программа должна вызывать эту функцию и выводить на экран результат ее работы.
2. Написать функцию, получающую в качестве аргумента имя файла, содержащего некоторую двоичную информацию, каждая запись которой представляется структурой `struct TRUNchar name[6];int ntrun;int lTime;`(название пункта назначения, номер поезда, время отправления) и возвращающую количество записей в файле. Функция должна возвращать —1, если она не смогла открыть файл. Основная программа должна сформировать файл, вызывать функцию и вывести на экран результат ее работы. Ввод данных произвести из программы, например `struct TRUNchar name[5];int ntrun;int lTime;myTrunc[]="tver",1,11,"Moscow",2,13,...`

За решение каждой задачи выставляется максимум 5 баллов.

Общая сумма В сумме за все задачи выставляет не более 40 баллов.

За работу на практических занятиях (решение задач у доски, выполнение домашних заданий) выставляется максимум 20 баллов.

За ответ на зачёте выставляется максимум 40 баллов.

Требования к рейтинг контролю (2 семестр)

Самостоятельная работа 1. Темы: алгоритмы сортировки и поиска, а именно сортировки вставками, сортировки пузырьком, сортировки Шелла. Пример задания:

1. Написать программу, сортирующую методом вставок главную диагональ двумерного массива целых чисел. Массив считать из текстового файла, содержащего в первой строке кол-во строк и кол-во столбцов, а далее в каждой строке целые числа, соответствующие элементам строки массива.
2. Написать программу, сортирующую методом вставок двумерный массив целых чисел.

При этом самое маленькое число должно оказаться в первом элементе первой строки, а самое большое в последнем элементе последней строки.

3. Написать программу, ищущую методом половинного деления строку с заданным значением 1-го элемента строки в двумерном массиве целых чисел. Выполнить все необходимые для этого операции. Не использовать стандартные функции.

За решение каждой задачи выставляется максимум 5 баллов.

Самостоятельная работа 2. Темы: динамические структуры данных, а именно линейные списки, односвязные и двусвязные списки. Пример задания:

1. Написать функцию, которая переворачивает список L, т.е. изменяет ссылки в этом списке так, чтобы его элементы оказались расположенными в обратном порядке.
2. Написать функцию сортировки по убыванию массива целых чисел, использующую двоичное дерево поиска. Результат записать в массив.

За решение каждой задачи выставляется максимум 5 баллов.

Самостоятельная работа 3. Темы: хеш-таблица, тестирование динамических структур данных.

Пример задания:

1. Для класса "группа студентов" построить хеш-таблицу по ключу – фамилия, написать функцию поиска в хеш-таблице номера записи по ключу, проверить работу хеш-поиска.
2. Протестировать алгоритм создания линейного списка, содержащего целые числа. Оценить время поиска элемента, время вставки элемента.
3. Протестировать алгоритм создания двоичного дерева поиска целых чисел. Оценить время поиска элемента, время вставки элемента.

За решение каждой задачи выставляется максимум 5 баллов.

Важной составляющей данного раздела РПД являются требования к рейтинг-контролю с указанием баллов, распределенных между модулями и видами работы обучающихся.

Максимальная сумма баллов по учебной дисциплине, заканчивающейся зачетом, по итогам семестра составляет 100 баллов (50 баллов - 1-й модуль и 50 баллов - 2-й модуль).

Студенту, набравшему 40 баллов и выше по итогам работы в семестре, в экзаменационной ведомости и зачетной книжке выставляется оценка «зачтено». Студент, набравший до 39 баллов включительно, сдает зачет.

Распределение баллов по модулям устанавливается преподавателем и может корректироваться.

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

Для аудиторной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 1л (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели
Помещение для самостоятельной работы обучающихся: Компьютерный класс № 4б 170002, Тверская обл., г.Тверь, Садовый переулок, д.35	Компьютер, экран, проектор, кондиционер.

Для самостоятельной работы

Наименование помещений	Материально-техническое оснащение помещений
Помещение для самостоятельной работы обучающихся: Компьютерный класс № 4б 170002, Тверская обл., г.Тверь, Садовый переулок, д.35	Компьютер, экран, проектор, кондиционер.
Компьютерная лаборатория факультета ПМиК № 201а	Набор учебной мебели, доска маркерная, компьютер, сервер (системный блок), онцентратор сетевой.

VIII. Сведения об обновлении рабочей программы дисциплины

№п.п.	Обновленный раздел рабочей программы дисциплины	Описание внесенных изменений	Реквизиты документа, утвердившего изменения
1	V. Учебно-методическое и информационное обеспечение дисциплины 2) Программное обеспечение	Внесены изменения в программное обеспечение	От 24.08.2023 года, протокол № 1 ученого совета факультета