

**Задачи для курса «Системное администрирование и программирование»
02.04.03 «Математическое обеспечение и администрирование информационных систем»**

Из каждого раздела выбрать минимум одну задачу (10 баллов).
Для большего числа баллов нужно выбрать еще максимум 4 задачи из разных разделов.

1. Упражнения для освоения синтаксиса языка Go по темам:

1. Реализовать функцию вычисления суммы элементов числового массива, переданного как параметр в эту функцию.
2. Реализовать функцию вычисления суммы положительных и сумму отрицательных элементов массива, переданного как параметр в эту функцию.
3. Для заданного отрезка $[a, b]$ построить массив значений функции $f(x)$. Реализовать алгоритм в виде отдельной функции.
4. Реализовать функцию вычисления k первых наименьших элементов числового массива.
5. Реализовать функцию вычисления k первых наибольших элементов числового массива.
6. Реализовать функцию дискретного преобразования Фурье сигнала, заданного одномерным числовым массивом.
7. Реализовать функцию вычисления медианного значения числового массива.
8. Реализовать функцию вычисления среднеквадратичного отклонения от среднего для заданного числового массива.
9. Реализовать процедуру вычисления элемента массива, для которого сумма модулей разностей с оставшимися элементами минимальна.
10. Реализовать функцию вычисления массива произведений всех двух соседних элементов заданного числового массива.
11. Реализовать функцию вычисления максимальной суммы подряд идущих положительных элементов заданного массива.
12. Реализовать процедуру вычисления суммы отрицательных элементов среза массива, заданного парой индексов.

2. Обработка символьных строк. Регулярные выражения.

1. Реализовать функцию, которая из заданного каталога выбирает имена файлов изображений.
2. Реализовать функцию, которая из указанного каталога строит ассоциативный массив, в котором ключами являются имена файлов, а значениями их размер.

3. Реализовать функцию, которая из указанного каталога строит ассоциативный массив, в котором ключами являются имена текстовых файлов, а значениями количество строк файла.
4. Реализовать функцию, которая из указанного каталога строит ассоциативный массив, в котором ключами являются имена текстовых файлов, а значениями количество слов файла.
5. Реализовать процедуру, которая из заданного текстового файла находит слова, состоящие из заданного количества букв.
6. Реализовать процедуру построения словаря по заданному текстовому файлу. Словарь представить как ассоциативный массив, ключами которого служат слова текста, а значениями – их частота в тексте.
7. Реализовать процедуру построения словаря по заданному текстовому файлу. Словарь представить в виде ассоциативного массива, ключами которого служат слова, а значениями – списки номеров предложений, в которых встречается это слово.
8. Реализовать процедуру поиска самого длинного слова в тексте.
9. Реализовать процедуру подсчета числа слов в тексте с заданным количеством букв.
10. Реализовать процедуру подсчета числа слов в тексте с заданным количеством гласных букв.
11. Реализовать процедуру подсчета числа слов в тексте с заданным количеством согласных букв.
12. Реализовать процедуру построения списка слов в тексте с заданным правилом чередования гласных и согласных букв. Это чередование должно быть задано словом как шаблоном.

3. Структуры. Методы.

1. Задать структуру для хранения в ней информации о шахматной партии, включая информацию об игроках, месте и времени игры. Реализовать методы добавления/удаления ходов, загрузку данных из внешнего файла, сохранение данных в файл, извлечение информации о ходе белых/черных с заданным номером, определение победителя, подсчет числа шахов, хороших и плохих ходов каждой стороны.
2. Задать структуру для хранения почтовых адресов, включая ФИО, город, индекс, улицу, номер дома и квартиры, а также тип строения, этажность, число проживающих, номер телефона. Реализовать методы добавления/удаления адреса, преобразование адреса в строку, извлечения номера телефона, города, индекса, подсчета числа адресов в заданном городе, загрузка и выгрузка всей структуры в файл.

3. Задать структуру данных для хранения электронных адресов включая ФИО, email, url страницы в одной из социальных сетей, возраст владельца. Реализовать методы добавления/удаления записей об электронных адресах, поиска по социальным сетям, по фамилии, загрузки и выгрузки данных в файл.
4. Определить структуру для хранения каталога книг личной библиотеки включая дерево тематического каталога, авторов, название книг, число страниц, тип переплета, наличие pdf версии. Реализовать методы добавления/удаления записей о книге, поиск по автору и названию, загрузки и выгрузки данных в файл.
5. Определить структуру для хранения данных о товарах магазина, включая название товара, категорию товара, количество товара, цену товара, место хранения. Реализовать методы добавления/удаления информации о товаре, методы поиска по названию и категории, поиск товара по диапазону цен в данной категории товаров, загрузка/выгрузка данных в файл.
6. Определить структуру для хранения данных о спортивных соревнованиях (к примеру по футболу), включая время, место проведения встречи, названия команд, их составы, результат матча. Реализовать методы добавления/удаления записей о матчах, выгрузку загрузку данных из файла, метод построения турнирной таблицы, сортированной по количеству набранных очков.
7. Определить структуру данных для хранения расписания занятий, включая наименования групп, названия дисциплин, аудиторий и времени занятий. Реализовать методы добавления/удаления записей о занятиях, загрузки/выгрузки данных в файл, метод проверки поиска накладок (два занятия в одно время и в одной аудитории, два занятия в одно время у одного преподавателя).
8. Определить структуру для хранения данных о театральных представлениях, включая время и дату, название пьесы, автор, жанр (драма, трагедия, водевиль, комедия, и т.п.), цена билета. Реализовать методы добавления/удаления записей, методы загрузки/выгрузки в файл, методы переноса представления с одной даты на другую, поиска по жанру, автору, названию.

4. Пакеты, Интерфейсы

1. Создать пакет `geometry` предоставляющий интерфейс для вычислений параметров геометрических фигур.
2. Создать пакет `polynom` предоставляющий интерфейс для символьных вычислений с полиномами.
3. Создать пакет `xdata` предоставляющий интерфейс для вычислений на последовательностях (вычисление средних значений, минимальных и максимальных значений, медианных значений, средне геометрических значений и т. п.)
4. Создать пакет `matrix`, предоставляющий интерфейс для матричных операций (транспонирования, произведения, умножения на вектор).
5. Создать пакет `triangle`, предоставляющий интерфейс для вычисления параметров треугольников. Реализовать процедуру определения попадает ли заданная точка в заданный треугольник.
6. Создать пакет `user`, предоставляющий интерфейс для управления учетными записями в ОС UNIX.
7. Создать пакет `student`, предоставляющий интерфейс для управления успеваемостью студентов.

5. Параллельное программирование. Горутины. Потоки и файлы

1. Реализовать параллельное вычисление максимального и минимального элементов одномерного массива.
2. Реализовать параллельное вычисление скалярного произведения векторов.
3. Реализовать параллельное вычисление медианного значения числовой последовательности.

4. Реализовать параллельное вычисление произведения матрицы на вектор.
5. Реализовать параллельное вычисление нормы квадратной матрицы. Норму выбрать по своему усмотрению.
6. Реализовать параллельный поиск в бинарном дереве по целочисленному ключу.
7. Реализовать параллельное вычисление количества слов в заданном тексте.
8. Реализовать параллельное вычисление слова максимальной длины в заданном тексте.
9. Реализовать параллельный поиск минимального положительного элемента числовой последовательности.

6. Обработка дерева каталогов файлов. Поиск и преобразование содержимого файлов.

В следующих задачах входным параметром является путь к каталогу. Необходимо выполнить обработку файлов во всех подкаталогах по всему дереву каталогов, начиная с заданного.

1. Найти файл максимального размера.
2. Построить список имен файлов, размер которых находится в заданном диапазоне.
3. Построить список имен файлов, название которых начинается на заданную букву.
4. Построить список имен файлов, дата создания которых раньше, чем заданная.
5. Построить список имен файлов, в которых имеется заданное слово.
6. Найти подкаталог, не имеющий подкаталогов, в котором сумма размеров файлов максимальная.
7. Найти подкаталог, не имеющий подкаталогов, в котором количество файлов максимально.
8. Построить список файлов с заданным «расширением».
9. Построить список файлов, принадлежащих заданному пользователю.
10. Построить список имен файлов, имеющих установленным бит SUID.
11. Построить список имен файлов «только для чтения».
12. Построить список имен файлов начинающихся с точки.