

Задания к работе №1 по прикладному программированию.

Все задания выполняются на языке программирования C (стандарт C99 и выше).

Реализованные приложения не должны завершаться аварийно.

Все ошибки, связанные с операциями открытия файла, должны быть обработаны; все открытые файлы должны быть закрыты.

Во всех заданиях запрещено использование глобальных переменных.

Во всех заданиях сравнение (на предмет эквивалентности или отношения порядка) вещественных чисел на уровне функции должно использовать значение эпсилон, которое (предпочтительнее) является параметром этой функции, либо (на худой конец) является значением, определённым глобально.

Во всех заданиях при пользовательском вводе с консоли необходимо организовать приглашение пользователя ко вводу в соответствии с типом и бизнес-логикой запрашиваемых данных.

Для всех реализованных алгоритмов необходимо уметь объяснить их асимптотические сложности.

1. Реализовать приложение, запрашивающее у пользователя три целых неотрицательных числа и проверяющее, могут ли эти три числа являться: а) сторонами треугольника; б) сторонами прямоугольного треугольника; в) элементами арифметической прогрессии; г) элементами геометрической прогрессии; д) элементами последовательности Фибоначчи; е) подряд идущими элементами последовательности Фибоначчи.
2. Реализовать приложение, запрашивающее у пользователя натуральное число, и проверяющее, является ли это число простым; является ли оно составным.
3. Реализовать функцию, строящую таблицу степеней (от 1 до 10) натуральных чисел от 1 до 6. При вычислениях запрещается использовать функции для возведения в степень (pow и др.). Продemonстрировать работу функции.
4. Реализовать функцию, меняющую местами значения двух чисел типа *int*. Входными параметрами функции являются указатели типа *int **. Продemonстрировать работу функции.
5. Реализовать функцию быстрого возведения целого числа в целую неотрицательную степень. Продemonстрировать работу функции.
6. Заполнить массив фиксированного размера BUFSIZ псевдослучайными целыми числами в диапазоне $[-10000..10000]$. Реализовать алгоритм поиска максимального и минимального элементов массива, а также индексов этих элементов, за один проход по массиву.
7. Реализовать функцию, которая принимает в качестве параметра строку и поднимает регистр всех гласных букв, находящихся на чётных позициях, а также опускает регистр всех согласных букв, находящихся на позициях, номера которых являются простыми числами (используйте функционал из задания 2). Продemonстрировать работу функции.
8. Реализовать функцию, которая принимает в качестве параметра строку и “переворачивает” (reverse) входную строку в той же области памяти. Продemonстрировать работу функции.
9. Реализовать функцию, которая принимает в качестве параметра строку и проверяет, является ли заданная строка палиндромом. Продemonстрировать работу функции.

10. Реализовать функцию конкатенации строк, аналогичную стандартной функции *strcat*.
Продemonстрировать работу функции.
11. Реализовать функцию сравнения строк, аналогичную стандартной функции *strcmp*.
Продemonстрировать работу функции.
12. Реализовать функцию преобразования целого числа типа *int*, заданного в системе счисления с основанием 10, в строковое представление этого числа в системе счисления с основанием из диапазона [2..36]. Продemonстрировать работу функции.
13. Реализовать функцию преобразования строкового представления числа в системе счисления с основанием из диапазона [2..36] в значение этого числа в типе *int*, заданного в системе счисления с основанием 10. Продemonстрировать работу функции.
14. В текстовом файле, путь к которому является аргументом командной строки, находятся строковые представления чисел, записанных в различных системах счисления в диапазоне [2..36], при этом информация о конкретной системе счисления для каждого числа утеряна. В файле строковые представления чисел разделены произвольным количеством разделителей (разделителями могут являться символы пробела, табуляции, переноса строки). Приложение должно прочитать файл, для каждого числа из файла определить минимальное основание системы счисления в диапазоне [2..36], в которой представление этого числа корректно, и в выходной файл (путь к файлу является аргументом командной строки) вывести число, определённую для него минимальное основание системы счисления и представление этого числа в десятичной системе счисления.
15. Реализовать в виде функций следующие алгоритмы сортировки: пузырьком, выбором, вставками, пирамидальная. Семантика функций и бизнес-контекст параметров реализованных функций должны соответствовать стандартной функции *qsort*.
Продemonстрировать работу функций.
16. Задана строка символов. Используя функции из задания 12, переписать строку так, чтобы в начале строки были символы цифр, затем символы букв латинского алфавита, а в самом конце – прочие символы; относительный порядок следования цифр/букв латинского алфавита/прочих символов при этом сохранить.
17. В текстовом файле записан целочисленный массив, размер которого не превосходит 128. Разделителем между строковыми представлениями чисел в файле являются символы пробела, табуляции, переноса строки. Файл корректен. Приложение должно прочитать файл, сохранить значения чисел из файла в массив и:
а) переписать в новый массив числа, стоящие на позициях, являющихся простыми числами;
б) переписать в новый массив нечётные числа;
в) найти среднее арифметическое элементов массива;
г) найти среднее геометрическое элементов массива;
д) найти среднее гармоническое элементов массива;
е) найти среднее арифметическое элементов массива;
ж) размах элементов массива;
з) разность сумм простых элементов и чётных положительных элементов массива.
18. Дан массив значений типа *double*. Скопировать его в новый массив, в котором должны отсутствовать дубликаты элементов из исходного массива.

- 19.Реализовать функции для: а) умножения матрицы на число; б) нахождения определителя матрицы; с) нахождения обратной матрицы. В случае невозможности выполнить операцию, необходимо сообщить пользователю об ошибке. Передачу матрицы как параметра функций реализовать при помощи передачи указателя на массив строк матрицы и её размеров (количество строк, количество столбцов). Продемонстрировать работу функций.
- 20.Реализовать функции для: а) сложения матриц; б) умножения матриц. Передачу матрицы как параметра функций реализовать при помощи передачи указателя на массив строк матрицы и её размеров (количество строк, количество столбцов). Продемонстрировать работу функции.
- 21.В текстовом файле записана целочисленная матрица, размеры которой не превосходят 100×100 . Разделителем между элементами строки матрицы являются символы пробела и табуляции; разделителем между строками матрицы - символ переноса строки. Файл является корректным. Приложение должно считать содержимое файла, заполнить матрицу и: а) вычислить ранг матрицы; б) найти все седловые точки матрицы (элемент матрицы $m[i][j]$ является седловой точкой тогда и только тогда, когда он является строго наименьшим в своей строке и строго наибольшим в своём столбце); с) вычислить след матрицы. В случае невозможности выполнить операцию, необходимо сообщить пользователю об ошибке.