

Задачи для 1 курса, часть 2

1. Простейшие вычислительные задачи

Упр.1. Для заданного N вычислить суммы в прямом и обратном порядке. При каких N получается максимальное расхождение в этих двух значениях?

$$\sum_{k=1}^N \frac{1}{\sqrt{k}}, \quad \sum_{k=1}^N \frac{\ln k}{k}, \quad \sum_{k=1}^N \frac{1}{k}, \quad \sum_{k=1}^N \frac{1}{k^2}, \quad \sum_{k=1}^N \frac{1}{k!}$$

Что изменится, если слагаемые умножить еще на $(-1)^k$?

Упр.2. Написать программу, определяющую машинную точность, т.е. максимальное $a > 0$, такое что $1 + a = 1$. Усложненный вариант: найти два ближайших к единице представимых вещественных числа.

В следующих задачах требуется оформить решение в виде функции, получающей в качестве параметра указатель на функцию `double (*f)(double x)`

1. Вычислить корень уравнения $f(x) = 0$ на данном отрезке с заданной точностью ϵ

- 1.1. методом деления пополам;
- 1.2. методом Ньютона (касательных);
- 1.3. методом хорд (секущих).

Написать тест, в котором вывести количество итераций и сравнить результат с точным ответом.

2. Вычислить интеграл от данной функции $f(x)$ на данном отрезке $[a, b]$ по составной формуле

- 2.1. прямоугольников с N отрезками;
- 2.2. трапеций с N отрезками;
- 2.3. Симпсона с N отрезками;
- 2.4. Гаусса (двухточечной) с N отрезками.

Сравнить с точным ответом.

3. Вычислить минимальное значение данной функции $f(x)$ с заданной точностью ϵ методом построения параболы с поиском трех точек с помощью

- 3.1. измельчения геометрической прогрессии;
- 3.2. метода золотого сечения;
- 3.3. измельчением равномерной сетки.

4. Написать программу вычисления значения одной из элементарных функций (`sin`, `cos`, `exp`, `log`) в заданной точке x и с заданной точностью ϵ суммированием ряда Тейлора. Сравнить с точным ответом

5. Написать программу, которая по значениям $x_1 \dots x_n$ и $y_1 \dots y_n$ вычисляет в данной точке x приближенное значение функции с помощью

- 5.1. интерполяционного многочлена Лагранжа;
- 5.2. кусочно-линейной интерполяции;
- 5.3. кусочно-квадратичной интерполяции;
- 5.4. Эрмитовой интерполяции (по $f(x_i)$ и $f'(x_i)$);
- 5.5. наилучшего среднеквадратического приближения линейной функцией.

Сравнить с точными значениями функции.

2. Задачи на работу с матрицами и метод Гаусса

Программы решения этих задач должны состоять из двух файлов: файл с функциями, решающими поставленную задачу, и файл с функциями формирования матриц и вывода ответа.

Формирование данных задачи (матрицы, правой части системы) выполняется: а) вводом из файла; б) вычислением по заданным формулам.

Каждую задачу из этого раздела нужно выполнить в двух вариантах представления матрицы:

а) как единого массива для всех элементов матрицы (функция получает аргумент `double *matr`)

б) как набора массивов-строк с массивом указателей на каждую строку. (функция получает аргумент `double **matr`)

0. Реализовать функции умножения матрицы на вектор, умножения двух прямоугольных матриц, функции для создания и вывода матриц.

1. Определить ранг вещественной $N \times M$ матрицы.
2. Найти определитель вещественной $N \times N$ матрицы.
3. Для квадратной матрицы вычислить обратную.
4. Решить систему линейных уравнений.

3. Обработка текстового файла

Программа должна состоять, как минимум, из двух файлов. В следующих задачах "словом" называется последовательность символов в пределах одной строки, не содержащая символов из заранее заданного набора (например, `“,;:;!?”()` и т.п.) При решении задач рекомендуется пользоваться стандартными функциями `strcmp`, `strcpy`, `strstr`, `strcat`, `strtok` и др. (см. `string.h`). Результатом работы программы должен быть новый, преобразованный файл.

0. Реализовать стандартные функции работы со строками `strcmp`, `strcpy`, `strstr`, `strcat`, `strset` и др.

1. Заменить в файле каждую последовательность заданных одинаковых символов на один такой символ.

2. Заменить всюду в файле один заданный набор символов на другой (с учетом разницы в их длине).

3. Вывести все слова из данного файла в другой файл в порядке их появления по одному слову на строке.

4. Вывести номера строк исходного файла и номер позиции в строке, где встречается заданное слово.

5. Определить максимальную, минимальную и среднюю длину слов из данного файла а также частоту (процент) появления каждого символа.

6. Вывести все слова из данного файла в алфавитном порядке.

7. Разрезать "длинные" строки в файле по пробелам на более короткие (не более заданной длины).

8. Удалить из файла часть текста между двумя "скобками", где скобка — это заданный набор символов. Например, убрать из файла комментарии в стиле C.

9. Реализовать инструкцию типа `#include`, т.е. вставить содержимое файла `filename` в то место файла, где встречается строка `#include filename`.

10. Реализовать инструкции типа `#define` и `#undef`, т.е. выполнить указанные подстановки в области их задания.

11. Реализовать инструкции типа `#ifdef` - `#else` - `#endif`, т.е. оставить в файле требуемый текст в зависимости от условия.

12. Отформатировать абзацы текста в заданных границах и с красной строкой (без переноса слов). Абзац — фрагмент текста между пустыми строками.

4. Итоговые задания.

1. Для заданной точности ϵ и различных значений параметра α найти все решения нелинейного уравнения, содержащего интегралы с особенностями. Например,

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt = \alpha.$$

2. Реализовать программу обработки множества файлов, заданных шаблоном или относящихся к указанному каталогу. Входные параметры должны задаваться в командной строке.

Конкретные варианты этих заданий будут даны в отдельном списке.

Минимальный набор заданий: по одному варианту каждой из задач 1, 2, 3, 4, 5 раздела 1, одна задача раздела 2, две задачи раздела 3, обе задачи раздела 4.

Сроки сдачи: разделы 1,2 — 31.03; раздел 3 — 20.04; раздел 4 — до 19.05.

Задачи для 1 курса, варианты раздела 4

1. Нелинейные уравнения с интегралами.

Эти уравнения могут быть решены одним из алгоритмов задачи 1.1. При этом для вычисления интеграла нужно реализовать один из методов интегрирования (задача 1.2.) с автоматическим выбором шага. Алгоритм решения следует реализовать так, чтобы не вычислять многократно интегралы по одним и тем же отрезкам (надо воспользоваться аддитивностью определенного интеграла). Отдельного исследования требует выбор точности вычисления интегралов и точности алгоритма решения уравнения. Для устранения особенности в интеграле нужно заменить подинтегральную функцию в окрестности особой точки на асимптотику, для которой интеграл вычисляется явно, и оценить полученную погрешность.

$$1. \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt = \alpha.$$

$$2. \int_0^x \frac{dt}{\sqrt{1+t^4}} = 1 + 0.1\alpha$$

$$3. \int_0^1 \sqrt{1+xt^4} dt = \alpha^2$$

$$4. \int_0^x \sqrt{t^3 + \frac{1}{t}} dt = \alpha^2 + 5\sqrt{x}$$

$$5. \int_0^x \sqrt{1 + \alpha^2 t^4} dt = \alpha + x$$

$$6. \alpha \int_0^x \frac{dt}{t\sqrt{1+t^4}} = \int_0^x \frac{dt}{\sqrt{1+t^4}}$$

$$7. \int_0^x \sqrt{\frac{1+t^3}{10+t^2}} dt = \alpha + x$$

$$8. \int_1^x \sqrt{t + \frac{1}{1+t^2}} dt = \alpha x$$

$$9. \int_{-x}^x \sqrt{1+t^5} dt = 3 - \frac{\alpha}{3}$$

$$10. \int_{-\infty}^x \frac{dt}{\sqrt{1+t^4}} = \alpha x^3$$

$$11. \int_0^x \frac{x-t}{\sqrt{1+t^4}} dt = \alpha$$

$$12. \int_1^x \int_0^1 \frac{dt ds}{\sqrt{t^4 + s^4}} = \alpha$$

2. Обработка группы файлов.

В этих задачах требуется освоить ввод данных в программу с помощью командной строки и познакомиться с операциями получения информации о файлах и каталогах. Для обхода каталогов наиболее удобно использовать функцию `ftw` (см. `man 3 ftw`).

1. Для заданного каталога и его подкаталогов определить все файлы, содержащие заданное слово.

2. Для заданного каталога и его подкаталогов напечатать таблицу имен файлов и их длины.

3. Для двух заданных каталогов напечатать список файлов, имеющих совпадающие имена.

4. Для двух заданных каталогов напечатать два списка файлов: присутствующих в одном каталоге и отсутствующих в другом.

5. Для заданного каталога и его подкаталогов напечатать список имен файлов, упорядоченных по размеру файла.

6. Для заданного каталога и его подкаталогов напечатать список файлов вместе с количеством строк в этих файлах (количество строк = количество символов `\n`).

7. Для заданного каталога и его подкаталогов найти все файлы, имеющие заданное расширение, и составить их список с полными путями.

8. Для заданного каталога и его подкаталогов вывести список всех файлов в виде дерева.

9. Для заданного каталога и его подкаталогов реализовать процедуру переименования всех файлов с данным расширением путем замены этого расширения на другое.

10. Для заданного каталога и его подкаталогов реализовать процедуру копирования всех файлов с данным расширением в другой указанный каталог.