**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ
ХАНТЫ-МАНСИЙСКОГО АВТОНОМНОГО ОКРУГА**

**СУРГУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ХМАО-ЮГРА**

Кафедра прикладной математики

Д. А. Моргун, А. Г. Назин

Вычислительная математика

Методические указания по выполнению практических работ для студентов факультета информационных технологий специальностей 010200(010501) «Прикладная математика и информатика» и 654600(230102)«Автоматизированные системы обработки информации и управления»

Часть 2

Сургут — 2006

Вычислительная математика : Методические указания по выполнению практических работ для студентов факультета информационных технологий специальностей 010200(010501) «Прикладная математика и информатика» и 654600(230102)«Автоматизированные системы обработки информации и управления». Часть 2/ Сост.: доцент. Д. А. Моргун, А. Г. Назин. — Сургут: Изд-во «Авиаграфия», 2006. — 32 с.

Методические рекомендации представляют собой варианты заданий, а также требования и рекомендации по оформлению практических работ.

Предназначены для студентов факультета информационных технологий, изучающих дисциплины «Численные методы» и «Вычислительная математика»

Библиогр.: 10 назв.

Печатается по решению редакционно-издательского совета Сургутского государственного университета.

Рецензент: А. В. Гореликов, канд. ф.-м. наук, доцент кафедры прикладной математики.

Содержание

1 Указания по выполнению и защите практических заданий

Практические задания даются студентам со следующими целями:

- Закрепление на практике теоретических знаний, полученных в курсе лекций по предметам «Численные методы», «Вычислительная математика»;
- Выработка и совершенствование навыков написания прикладных программ на языках программирования `Pascal` или `Delphi`;
- Проверка и оценка полученных студентами знаний и навыков.

Постановку задачи (т.е., функцию, уравнение, систему уравнений, матрицу и т.п.), а также ***метод решения*** преподаватель задаёт каждому студенту индивидуально.

Для того, чтобы практическая работа была зачтена, студенту необходимо:

1. Самостоятельно написать и отладить программу.
2. Ответить на вопросы преподавателя (по постановке задачи, методу решения, листингу программы).
3. По требованию преподавателя, внести небольшие изменения[1] в программу (изменить функцию, уравнение или систему уравнений, поменять начальное приближение, вычислить результат с большей или меньшей точностью).

Если студент не отвечает на дополнительные вопросы, или отвечает неправильно, или неспособен внести требуемые изменения в программу, но при этом программа работает правильно, то это расценивается, как работа, выполненная ***несамостоятельно***. Такая работа получает оценку ***«незачёт»***, а студент получает другое задание на ту же тему.

На стр.31 приведён список рекомендуемой литературы, имеющейся в библиотеке СурГУ.

[1] Если программа реализована грамотно, то изменения действительно будут небольшими и будут заключаться в изменении значений или констант, а также в переопределении функций. Неграмотно написанную программу, возможно, придётся переписывать полностью.

1.1 Листинг программы

Стиль программирования. К третьему курсу (а предметы «Численные методы» и «Вычислительная математика» преподаются, обычно, не раньше третьего курса) у успевающих студентов, как правило, уже есть сформировавшийся собственный стиль написания программ. И многим студентам непонятно, почему они должны подстраивать свой стиль оформления под какие-либо общие требования. К сожалению, зачастую, стили разных студентов существенно отличаются между собой и от общепризнанных правил оформления листинга.

Конечно же, **главное** требование — правильная работа программы. В случае, если программой занимается только один программист и от него требуется только готовый откомпилированный результат, — это требование является достаточным.

Но работа прикладного математика или программиста не обязательно индивидуальна. Сложный, трудоёмкий проект может охватывать различные области науки и требовать взаимодействия различных специалистов. Большие проекты являются результатом совместного труда многих специалистов. В таких проектах очень важно взаимодействие между отдельными исполнителями, умение грамотно оформить техническую документацию (листинг программ относится к категории технической документации).

Для облегчения взаимопонимания специалистов, необходимо оформлять техническую документацию в соответствии с некоторыми требованиями. Это могут быть требования, утверждённые государством (ГОСТы), утверждённые руководством фирмы, или просто устоявшиеся традиции.

Как бы то ни было, ***специалист должен уметь оформлять листинг своих программ в соответствии с требованиями.*** Для закрепления этого умения, а также для упрощения проверки студенческих программ и оказания консультаций по отладке, студенты должны выполнять «**Требования к оформлению листинга программы**» (см. Приложение 1).

Программы, листинг которых оформлен не по правилам, будут возвращаться на доработку.

1.2 Интерфейс программы

Ввод

Как показывает опыт, специфика данного предмета такова, что на отладку программ тратится значительно больше времени, чем на получение тестовых результатов и на процесс защиты программы. Поэтому, в целях экономии времени, следует ограничить или полностью исключить ввод параметров задачи с клавиатуры. Все величины, характеризующие рассматриваемую задачу, нужно задавать в разделе констант (`const`) программы.

Ввод векторных или матричных величин необходимо производить из текстовых файлов.

Перед написанием программы необходимо выделить все параметры задачи, которые являются постоянными во время одного запуска программы, но могут измениться во время следующего (Например, в программе для построения графика такими параметрами являются масштабные коэффициенты и величины смещения вдоль осей). Все подобные величины необходимо сгруппировать в раздел констант и вынести этот раздел в начало листинга. Таким образом, все параметры, описывающие задачу и регулирующие работу программы можно будет очень быстро найти.

Вывод

Ниже приведены элементы, которые программа должна выводить на экран в процессе работы.

Постановка задачи: Сюда входит: название и тип задачи, если есть; конкретное уравнение или система, которое решается; метод решения; числовые характеристики метода, если есть.

Процесс решения: Здесь приводится необходимая техническая информация (если нужно). Например, номер итерации и величина невязки.

Полученный результат: Получив решение, его нужно вывести на экран. Но кроме решения, обычно нужно вывести проверку, невязку, количество итераций, и, возможно, другую информацию, в зависимости от специфики задачи и метода решения. **Все числовые значения, выводимые на экран, должны быть подписаны**.

2 Построение графиков

Задание одинаковое для всех. Необходимо написать программу, которая может строить графики функции одной переменной, как аналитически заданной, так и таблично заданной.

Пример аналитически заданной функции: $f(x) = \sin x$

Пример таблично заданной функции:

x_i	0,1	0,2	...	1,0
$f(x_i)$	1,0	3,0	...	5,3

Для таблично заданной функции необходимо организовать чтение исходных данных из текстового файла.

Общие требования:

Оси координат. Изображаемый на экране центр системы координат должен соответствовать реальному центру. На осях должны быть нанесены деления, размер которых позволит визаульно определять координаты интересующих нас точек графика

Масштабирование. Необходимо задать масштабирующие множители, которые позволят сжимать или растягивать график вдоль оси OX и вдоль оси OY

Смещение центра координат. Необходимо ввести отдельные переменные (или константы) для координат центра. Изменение этих величин позволит перемещать центр координат на экране и более подробно рассматривать интересующую нас область экрана.

3 Нелинейные уравнения и системы

3.1 Уравнения с одним неизвестным

Каждому студенту индивидуально определяется нелинейная функция $f(x)$ и задаётся метод решения. Необходимо найти любой корень уравнения $f(x) = 0$ с точностью $\varepsilon = 10^{-4}$.

Этапы выполнения данного задания:

Локализация корней. Поскольку $f(x)$ — нелинейная функция, то она может иметь конечное или бесконечное число корней, или не иметь их вовсе. Все приведённые ниже задания сформулированы для итерационных методов. Таким методам для начала расчётов нужно задать начальное приближение вблизи корня. Если приближение задано вдали от корня, то метод может расходиться или сходиться не к тому корню, который нас интересует. Если корней не существует вовсе, то применять численные методы бессмысленно. Для того, чтобы выяснить, существуют ли корни, и локализовать[2] их, проще всего использовать графический метод. Для построения графика нужно использовать программу из пункта 2.

Решение уравнения. Нужно написать программу, реализующую заданный численный метод.

Проверка. Задав начальное приближение, запустить программу и получить результат. Программа должна выводить на экран решение $x^{(k)}$, количество итераций k, значение $f(x^{(k)})$ и точность ε

Методы решения:

1. Метод дихотомии
2. Метод простых итераций
3. Метод Ньютона, непрерывный[3] вариант
4. Метод Ньютона, разностный[4] аналог
5. Метод секущих (хорд)
6. Метод парабол, первый[5] вариант
7. Метод парабол, второй[6] вариант

[2]Локализовать — т.е., определить промежутки, на которых лежит строго по одному корню. Чем более узкие границы отрезка локализации будут выбраны, тем быстрее сойдётся численный метод. Кроме того, на отрезке локализации функция должна вести себя монотонно

[3]Непрерывный вариант метода Ньютона есть просто оригинальный вариант формулы, в которой производная в знаменателе вычисляется аналитически

[4]В разностном, или дискретном аналоге, производная в знаменателе заменяется разностным выражением

[5]Уравнение параболы строится по значениям функции в трёх точках $f(x^{(k)})$, $f(x^{(k-1)})$ и $f(x^{(k-2)})$

[6]Уравнение параболы строится с помощью ряда Тейлора, т.е. по $f(x^{(k)})$, $f'(x^{(k)})$ и $f''(x^{(k)})$

Варианты функций:

1. $f(x) = \sin x + x^2 - \frac{x}{20}$
2. $f(x) = x^{-x} + x^2 - 3$
3. $f(x) = \frac{x^3}{8} + x^2 - \cos 2x$
4. $f(x) = -x^3 + 4x + \cos x$
5. $f(x) = x^3 + 2x^2 - 8$
6. $f(x) = x^2 - \cos\left(\sqrt[3]{x}\right)$
7. $f(x) = e^{-x^2} - \cos x^2 - 2$
8. $f(x) = e^{-x^2} - \sin x$
9. $f(x) = \cos\sqrt{|x|} - x$
10. $f(x) = \cos x + \frac{(x-3)^2}{2} - \frac{1}{2}$
11. $f(x) = x - \sin\left(\sqrt[3]{x}\right)$
12. $f(x) = e^x + e^{-x} - x$
13. $f(x) = x^5 - 8x^3 + 2x^2 + x - 2$
14. $f(x) = x + 2\sin x + \cos 3x$
15. $f(x) = \frac{x^2}{5} - \sin\left(\sqrt[3]{x}\right) - 5$
16. $f(x) = x^3 - 3x^2 + 8$
17. $f(x) = \cos(1{,}7^x - 1)$
18. $f(x) = \sin(1{,}3^x - 1{,}2)$
19. $f(x) = \exp(\sin x) - x$
20. $f(x) = \sin x - \cos\left(x^{1{,}5}\right)$
21. $f(x) = \sin\left(x^{1{,}3}\right) + \cos x$
22. $f(x) = \ln x - \operatorname{tg} x$
23. $f(x) = \ln(x + 2{,}5) - x^2 + 1$
24. $f(x) = x\cos x + \sin x$
25. $f(x) = x\sin x + \cos x$
26. $f(x) = e^{-x} - \sin x$
27. $f(x) = \ln x - 4 + x^2$
28. $f(x) = \sqrt{x} + x^2 - 10$
29. $f(x) = x^5 + 2x - 8$
30. $f(x) = \sqrt{x} - \cos x$
31. $f(x) = e^{-x^2} - x^2 + 2x$
32. $f(x) = e^x - \frac{1}{\sqrt{x}}$
33. $f(x) = \sqrt{\cos x} - x^2$
34. $f(x) = \ln(2 + x) - \sqrt{x}$
35. $f(x) = \ln(x + 1) - \sqrt{4 - x^2}$
36. $f(x) = \sqrt{x + 1} - \sqrt{x} - 0{,}5$
37. $f(x) = 2^x - 2 + x^2$
38. $f(x) = \sin x - \frac{1}{(1+x^2)}$
39. $f(x) = \ln\left(1 + \sqrt{x}\right) - \cos x$
40. $f(x) = x^2 - \operatorname{ctg} x$

3.2 Системы нелинейных уравнений

Необходимо решить систему нелинейных уравнений $\vec{F}(\vec{x}) = 0$, где

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \vec{F} = \begin{pmatrix} f_1(\vec{x}) \\ f_2(\vec{x}) \\ \vdots \\ f_n(\vec{x}) \end{pmatrix}, \quad \left\{ \begin{array}{c} f_1(x_1, x_2, \ldots, x_n) = 0 \\ f_2(x_1, x_2, \ldots, x_n) = 0 \\ \ldots \\ f_n(x_1, x_2, \ldots, x_n) = 0 \end{array} \right. \quad \Leftrightarrow \quad \vec{F}(\vec{x}) = 0$$

Данное задание аналогично предыдущему и отличается от него только многомерностью. Действительно, сравните два уравнения: $f(x) = 0$ для одномерного случая и $\vec{F}(\vec{x}) = 0$ для многомерного.

Этапы выполнения данного задания также аналогичны предыдущему случаю:

Локализация корней. Использовать графический метод здесь также удобнее всего, но возникают сложности, связанные с многомерностью. Если $n = 2$ (система из двух нелинейных уравнений, каждое уравнение зависит от двух неизвестных), то обе функции, входящие в систему, представляют собой поверхности в трёхмерном пространстве. Каждая из этих поверхностей пересекает координатную плоскость OXY по некоторой кривой (если не пересекает, то решения не существует). В свою очередь, точки пересечения таких кривых (если они существуют) являются решениями системы. Таким образом, для двумерного случая, графическую локализацию корней можно провести двумя способами:

1. Построить графики функций в трёхмерном пространстве и по ним локализовать корни. Достаточно универсальный способ, но самостоятельное программирование отображения трёхмерной графики может потребовать много времени. Поэтому для построения трёхмерных графиков допускается использование фирменных программных продуктов (`Golden Software Surfer`, `TecPlot`, `MathCAD` и др.)
2. Представить уравнения системы в виде $y = f(x)$ (1-е уравнение) и $y = g(x)$ (2-е уравнение). Такой переход не всегда осуществим. Но если функции достаточно просты и указанное преобразование удалось реализовать, то остаётся построить графики кривых $y = f(x)$ и $y = g(x)$, а точки их пересечения будут корнями системы.

Решение уравнения. Написать программу, реализующую заданный численный метод. В качестве критерия выхода использовать условие $\left\| \vec{x}^{(k+1)} - \vec{x}^{(k)} \right\|_E < \varepsilon$.

Проверка. Задав начальное приближение, запустить программу и получить результат. Программа должна выводить на экран решение $\vec{x}^{(k)}$, количество итераций k, вектор $\vec{F}(\vec{x}^{(k)})$ и точность ε.

Рекомендации. Для работы с векторными и матричными величинами разумно описать соответствующие процедуры и функции. Процедуры для произведения векторов, сложения, вычитания, для умножения матрицы на вектор; функции для скалярного произведения векторов, для умножения вектора на число, для вычисления нормы вектора и нормы матрицы (формулы для различных норм см. в Приложении 2). Кроме того, могут пригодиться процедуры для вывода векторов и матриц на экран.

Методы решения:

1. Неявный метод Ньютона
2. Явный метод Ньютона
3. Разностный аналог неявного метода Ньютона
4. Разностный аналог явного метода Ньютона
5. Модифицированный метод Ньютона
6. Метод простой итерации
7. Метод Брауна

Варианты систем:

1. $\begin{cases} \sin 0{,}2x_2 - \cos 0{,}4x_1 = -0{,}1 \\ e^{-((x_1-2)/4)^2} e^{-((x_2-1)/3)^2} = 0{,}1 \end{cases}$

2. $\begin{cases} 1{,}5^{x_1-5} - 0{,}1x_2 = 0 \\ e^{-((x_1-2)/4)^2} \cos(x_2/3) = 0{,}5 \end{cases}$

3. $\begin{cases} 1{,}1^{-(x_1^2 - x_1x_2 + 2x_2^2)} = 0{,}1 \\ \cos\left(0{,}03x_1^2 - 0{,}04x_1x_2 + 0{,}01x_2^2\right) - \\ \qquad - 0{,}1x_1x_2 = 0 \end{cases}$

4. $\begin{cases} \operatorname{arcctg}\left(x_1^2 + 2x_1x_2 + 2x_2 + 0{,}5\right) = 0{,}2 \\ \cos\left(0{,}1x_2^2 + x_2 - 0{,}02x_1^2\right) = 0 \end{cases}$

5. $\begin{cases} \operatorname{arctg}(-x_1\cos x_2) + x_1 = 0{,}2 \\ x_1x_2\sin\left(x_1 + x_2 - 2x_2^2\right) = 0{,}1 \end{cases}$

6. $\begin{cases} \operatorname{tg}(0{,}3\cos x_1) - \exp\left(-x_2^2\right) = -1 \\ \exp\left(-(x_1/4)^2\right) - \operatorname{tg}(0{,}2\sin x_2) = 0{,}6 \end{cases}$

7. $\begin{cases} \operatorname{tg}(0{,}3\cos x_1) - \operatorname{tg}(0{,}2\sin x_2) = -0{,}3 \\ \sin\left(0{,}1x_1^2 - 0{,}2x_2\right) = 0{,}01 \end{cases}$

8. $\begin{cases} \cos\left(\ln\left(0{,}5 + x_1^2\right)\right) - \\ \quad - \sin\left(\ln\left(0{,}4 + (x_2/2)^2\right)\right) = 0{,}025 \\ 4 \cdot \left(x_1^2 + 2x_2^2 + 4 - \right. \\ \qquad \left. - \cos(0{,}01x_1x_2)\right)^{-1} = 0{,}3 \end{cases}$

9. $\begin{cases} \cos(5\sin(0{,}1x_1) - 3\sin(0{,}3x_2)) = 0{,}1 \\ \sin(3\cos(0{,}3x_1) - 5\cos(0{,}1x_2)) = 0{,}3 \end{cases}$

10. $\begin{cases} \sinh(x_1 + 0{,}2x_2 + \operatorname{tg}(0{,}1x_1x_2)) = 0{,}8 \\ \sinh(0{,}6x_1 - 0{,}1x_2 + \operatorname{tg}(0{,}2x_1x_2)) = 0{,}1 \end{cases}$

4 Минимизация функций

4.1 Минимум функции 1 переменной.

Для заданной функции одной переменной $f(x)$ необходимо найти любой экстремум[7] с помощью метода, указанного преподавателем. Точность $\varepsilon = 10^{-4}$.

Этапы выполнения:

Локализация экстремумов. Графический метод.

Решение уравнения. Написать программу, реализующую заданный численный метод.

Проверка. Задав начальное приближение, запустить программу и получить результат. Программа должна выводить на экран решение $x^{(k)}$, количество итераций k, величины $f(x^{(k)})$, $f'(x^{(k)})$ и точность ε

Методы решения:

1. Метод золотого сечения
2. Метод Ньютона
3. Метод Ньютона, разностный аналог
4. Метод парабол, первый вариант[8]
5. Метод парабол, второй вариант

Варианты функций см. на стр.9.

[7]Минимум, максимум или экстремум любого типа — по указанию преподавателя.
[8]О различиях между первым и вторым вариантом метода парабол см. сноску на стр.8

4.2 Многомерная минимизация.

Для заданной функции двух переменных $\Phi(x,y)$ необходимо найти минимум с помощью метода, указанного преподавателем. Точность $\varepsilon = 10^{-4}$.

Этапы выполнения:

Локализация минимума. Графический метод[9].

Решение уравнения. Написать программу, реализующую заданный численный метод. В качестве критерия выхода использовать условие $\left\|\vec{x}^{(k+1)} - \vec{x}^{(k)}\right\|_E < \varepsilon$.

Проверка. Задав начальное приближение, запустить программу и получить результат. Программа должна выводить на экран решение $\vec{x}^{(k)}$, количество итераций k, величины $\Phi(\vec{x}^{(k)})$, $\nabla\Phi(\vec{x}^{(k)})$ и точность ε.

Методы решения:

1. Метод градиентного спуска
2. Метод покоординатного спуска
3. Метод наискорейшего спуска
4. Метод параллельных касательных плоскостей

Варианты функций. Из указанной преподавателем системы нелинейных уравнений (см. стр. 11) необходимо следующим образом сконструировать функцию $\Phi(\vec{x}^{(k)})$ для минимизации:

$$\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \begin{cases} f(x,y) = 0 \\ g(x,y) = 0 \end{cases} \quad \rightarrow \quad \Phi(x,y) = \Phi(\vec{x}) = f^2(\vec{x}) + g^2(\vec{x})$$

Если у выбранной системы нелинейных уравнений существует решение, то, очевидно, на этом же решении будет достигать минимума вышеприведённая функция $\Phi(\vec{x})$, причём в точке минимума $\Phi(\vec{x}_{min}) = 0$.

[9] См. аналогичный пункт темы «Системы нелинейных уравнений» на стр.10

5 Вычислительные задачи линейной алгебры

Вычислительные задачи линейной алгебры подразделяются на:

1. Решение систем линейных алгебраических уравнений (СЛАУ);
2. Нахождение определителей;
3. Нахождение обратных матриц;
4. Задачи на собственные значения.

Методы, применяемые для решения этих задач, подразделяются на ***прямые*** и ***итерационные***. В связи с тем, что многие прямые методы, предназначенные для решения СЛАУ, после небольшой модификации применимы также для нахождения определителей и обратных матриц, практические задания сгруппированы следующим образом:

1. Прямые методы для задач линейной алгебры.
2. Итерационные методы решения СЛАУ.
3. Задачи на собственные значения.

5.1 Прямые методы для задач линейной алгебры

Ниже будут представлены варианты заданий для прямых методов решения СЛАУ,

$$\mathbb{A}\vec{x} = \vec{b}; \quad \text{где} \quad \mathbb{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}; \quad \vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}; \quad \vec{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

а также для их модификаций, позволяющих находить определители

$$det(\mathbb{A}) = |\mathbb{A}|$$

и обратные матрицы $\mathbb{A}^{-1}$

Этапы выполнения:

Ввод данных из текстового файла. Необходимо реализовать процедуру, считывающую из текстового файла данные о задаче. Текстовый файл будет иметь следующий формат:

$$\begin{array}{lllll} n & & & & \\ a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ . & . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}$$

где n — размерность системы, a_{ij} — элементы матрицы $\mathbb{A}$, b_i — компоненты вектора-столбца свободных членов $\vec{b}$. Для задач на нахождение $det(\mathbb{A})$ или $\mathbb{A}^{-1}$ столбец свободных членов $\vec{b}$ должен **игнорироваться**.

Решение. Написать программу, реализующую заданный численный метод.

Проверка. Задав имя файла, содержащего данные о задаче, запустить программу и получить результат. Программа должна выводить на экран исходные данные[10], считанные из файла, решение[11], проверку[12].

Методы решения:

1. Нахождение $\mathbb{A}^{-1}$ методом Гаусса
2. Нахождение $\mathbb{A}^{-1}$ методом $\mathbb{LU}$-разложения
3. Решение СЛАУ методом Гаусса с постолбцовым выбором ведущего элемента
4. Решение СЛАУ методом $\mathbb{LU}$-разложения
5. Решение СЛАУ методом вращений
6. Решение СЛАУ методом квадратных корней
7. Нахождение $det(\mathbb{A})$ по методу $\mathbb{LU}$-разложения
8. Нахождение $det(\mathbb{A})$ по методу квадратных корней
9. Нахождение $det(\mathbb{A})$ по методу Гаусса с постолбцовым выбором ведущего элемента
10. Прогонка
11. Метод отражений

[10] Для задач на решение СЛАУ — матрицу $\mathbb{A}$ и вектор $\vec{b}$, для задач на нахождение определителя или обратной матрицы — только матрицу $\mathbb{A}$.

[11] Вектор $\vec{x}$ для задач на решение СЛАУ, матрицу $\mathbb{A}^{-1}$ для задач на обратную матрицу и значение $|\mathbb{A}|$ для задач на нахождение определителя.

[12] Т.е., невязку $\vec{\xi} = \vec{b} - \mathbb{A}\vec{x}$ для задач на СЛАУ; для задач на обратную матрицу — значение произведения матриц $\mathbb{A}\mathbb{A}^{-1}$. Для задачи на нахождение определителя в качестве проверки будет использоваться результат вычисления определителя по другому методу.

Варианты СЛАУ

1. $$\begin{pmatrix} -3{,}8 & 0{,}8 & 1{,}2 \\ -17{,}4 & 4{,}4 & 3{,}6 \\ -10{,}6 & 0{,}6 & 5{,}4 \end{pmatrix} \vec{x} = \begin{pmatrix} 16{,}6 \\ 35{,}8 \\ 88{,}2 \end{pmatrix}$$

2. $$\begin{pmatrix} -6{,}2 & 3{,}2 & -1{,}2 \\ -24{,}6 & 11{,}6 & -3{,}6 \\ -15{,}4 & 5{,}4 & 0{,}6 \end{pmatrix} \vec{x} = \begin{pmatrix} -2{,}2 \\ -10{,}6 \\ -5{,}4 \end{pmatrix}$$

3. $$\begin{pmatrix} -1{,}00 & 1{,}50 & 2{,}25 \\ -3{,}00 & 4{,}00 & 2{,}00 \\ -6{,}00 & 2{,}00 & 7{,}00 \end{pmatrix} \vec{x} = \begin{pmatrix} 10{,}5 \\ 10{,}0 \\ 2{,}0 \end{pmatrix}$$

4. $$\begin{pmatrix} 6{,}00 & -4{,}75 & 5{,}25 \\ 2{,}00 & 0{,}75 & 0{,}75 \\ 0{,}67 & -0{,}42 & 2{,}25 \end{pmatrix} \vec{x} = \begin{pmatrix} 70{,}75 \\ 23{,}25 \\ 16{,}44 \end{pmatrix}$$

5. $$\begin{pmatrix} 23{,}0 & -18{,}0 & -12{,}0 \\ 16{,}2 & -12{,}4 & -8{,}4 \\ 7{,}8 & -6{,}6 & -2{,}6 \end{pmatrix} \vec{x} = \begin{pmatrix} -40{,}0 \\ -26{,}4 \\ -13{,}6 \end{pmatrix}$$

6. $$\begin{pmatrix} 3 & 5 & -3 \\ 1 & 11 & -5 \\ 1 & 13 & -5 \end{pmatrix} \vec{x} = \begin{pmatrix} 64 \\ 84 \\ 100 \end{pmatrix}$$

7. $$\begin{pmatrix} -4{,}17 & 2{,}00 & 0{,}17 \\ -41{,}17 & 14{,}00 & 1{,}17 \\ -7{,}17 & 2{,}00 & 3{,}17 \end{pmatrix} \vec{x} = \begin{pmatrix} -19{,}66 \\ -209{,}66 \\ -13{,}66 \end{pmatrix}$$

8. $$\begin{pmatrix} 5{,}67 & -0{,}00 & -0{,}67 \\ 4{,}13 & 5{,}80 & -4{,}53 \\ 3{,}87 & 1{,}20 & 0{,}53 \end{pmatrix} \vec{x} = \begin{pmatrix} 56{,}03 \\ 123{,}77 \\ 57{,}23 \end{pmatrix}$$

9. $$\begin{pmatrix} 0{,}84 & 0{,}64 & 0{,}24 \\ -7{,}40 & 3{,}60 & 3{,}60 \\ -7{,}04 & 0{,}16 & 6{,}56 \end{pmatrix} \vec{x} = \begin{pmatrix} 14{,}4 \\ -60{,}0 \\ -70{,}4 \end{pmatrix}$$

10. $$\begin{pmatrix} 16{,}1250 & 1{,}0000 & -9{,}7500 \\ 8{,}4375 & 7{,}5000 & -10{,}1250 \\ 6{,}1875 & 1{,}5000 & -2{,}6250 \end{pmatrix} \vec{x} = \begin{pmatrix} 56{,}75 \\ 38{,}625 \\ 25{,}125 \end{pmatrix}$$

11. $$\begin{pmatrix} 2{,}57 & -2{,}57 & 3{,}29 & 0{,}14 \\ 0{,}00 & -1{,}00 & 4{,}00 & 0{,}00 \\ -1{,}00 & -4{,}00 & 8{,}00 & 1{,}00 \\ -0{,}29 & -7{,}71 & 4{,}86 & 6{,}43 \end{pmatrix} \vec{x} = \begin{pmatrix} 4{,}87 \\ 9{,}00 \\ 12{,}00 \\ -2{,}41 \end{pmatrix}$$

12. $$\begin{pmatrix} 5{,}25 & -2{,}75 & 3{,}25 & -0{,}50 \\ 1{,}00 & 2{,}00 & 3{,}00 & -2{,}00 \\ 2{,}75 & 1{,}75 & 6{,}75 & -5{,}50 \\ 0{,}00 & -4{,}00 & 2{,}00 & 5{,}00 \end{pmatrix} \vec{x} = \begin{pmatrix} 15{,}00 \\ 2{,}00 \\ -11{,}00 \\ 21{,}00 \end{pmatrix}$$

13. $$\begin{pmatrix} 1{,}4 & -0{,}5 & 2{,}0 & -0{,}3 & -0{,}6 & 0{,}8 \\ 0{,}0 & 2{,}0 & 0{,}0 & 0{,}0 & 0{,}0 & 0{,}0 \\ -3{,}4 & -2{,}0 & 6{,}3 & -1{,}0 & 0{,}3 & 3{,}0 \\ 1{,}3 & -3{,}0 & -0{,}9 & 5{,}0 & -0{,}9 & 0{,}5 \\ -1{,}0 & 1{,}0 & 1{,}5 & 0{,}5 & 2{,}5 & -1{,}7 \\ -0{,}8 & -2{,}4 & 1{,}4 & 0{,}3 & -1{,}1 & 4{,}1 \end{pmatrix} \vec{x} = \begin{pmatrix} 3{,}2 \\ 4{,}0 \\ 1{,}9 \\ -5{,}1 \\ 0{,}8 \\ -0{,}1 \end{pmatrix}$$

14. $$\begin{cases} -9{,}93x_1 + 7{,}21x_2 = -1{,}39 \\ 2{,}21x_1 - 9{,}93x_2 + 7{,}71x_3 = 1{,}48 \\ 1{,}71x_2 - 9{,}93x_3 + 8{,}21x_4 = 0{,}48 \\ 1{,}21x_3 - 9{,}93x_4 + 8{,}71x_5 = -0{,}85 \\ 0{,}71x_4 - 9{,}93x_5 + 9{,}21x_6 = -2{,}22 \\ 0{,}21x_5 - 9{,}93x_6 = -24{,}36 \end{cases}$$

15. $$\begin{cases} -16{,}41x_1 + 9{,}20x_2 = -6{,}88 \\ 6{,}20x_1 - 16{,}41x_2 + 10{,}20x_3 = 1{,}79 \\ 5{,}20x_2 - 16{,}41x_3 + 11{,}20x_4 = 3{,}25 \\ 4{,}20x_3 - 16{,}41x_4 + 12{,}20x_5 = 3{,}69 \\ 3{,}20x_4 - 16{,}41x_5 + 13{,}20x_6 = 1{,}93 \\ 2{,}20x_5 - 16{,}41x_6 + 14{,}20x_7 = -2{,}40 \\ 1{,}20x_6 - 16{,}41x_7 + 15{,}20x_8 = -8{,}01 \\ 0{,}20x_7 - 16{,}41x_8 = 4{,}11 \end{cases}$$

16. $$\begin{cases} 2x_1 - x_2 = 2 \\ -x_1 + 2x_2 - x_3 = 2 \\ -x_2 + 2x_3 - x_4 = 2 \\ -x_3 + 2x_4 = 17 \end{cases}$$

17. $$\begin{cases} 2{,}0x_1 - 0{,}5x_3 + 0{,}5x_5 = 0{,}0 \\ 5{,}3x_2 + 1{,}3x_4 - 2{,}7x_6 = 29{,}4 \\ -8{,}0x_1 + 5{,}0x_3 + 5{,}0x_5 = -8{,}0 \\ -2{,}7x_2 + 9{,}3x_4 - 2{,}7x_6 = 5{,}4 \\ -8{,}0x_1 + 4{,}0x_3 + 6{,}0x_5 = -12{,}0 \\ -2{,}7x_2 + 1{,}3x_4 + 5{,}3x_6 = -34{,}6 \end{cases}$$

5.2 Итерационные методы решения СЛАУ

Постановка задачи, этапы выполнения, требования к оформлению и контрольные примеры — такие же, как в предыдущем пункте.

Дополнительное требование: вывести на экран **количество итераций**.

Критерий выхода из итерационного цикла: $\left\|\vec{x}^{(k+1)} - \vec{x}^{(k)}\right\|_c < \varepsilon$, где $\varepsilon = 10^{-4}$.

Методы решения:

1. Метод Якоби
2. Метод Зейделя
3. Метод релаксаций

5.3 Алгебраическая проблема собственных значений

Необходимо найти λ и $\vec{x}$, удовлетворяющие условию:

$$\mathbb{A}\vec{x} = \lambda\vec{x}, \quad \text{где} \quad \mathbb{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Тривиальный случай $\vec{x} = 0$ не представляет практического интереса. Интерес представляют ненулевые *собственные значения* λ и соответствующие им *собственные вектора* $\vec{x}$ (в совокупности называемые парой $\{\lambda, \vec{x}\}$). Уравнение $\mathbb{A}\vec{x} = \lambda\vec{x}$ часто представляют в виде

$$(\mathbb{A} - \lambda\mathbb{E})\vec{x} = 0, \quad \text{где} \quad (\mathbb{A} - \lambda\mathbb{E}) = \begin{pmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{12} - \lambda & \dots & a_{2n} \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{pmatrix}$$

Вышеприведённое уравнение относится к однородным линейным системам и имеет нетривиальное решение при выполнении равенства

$$det(\mathbb{A} - \lambda\mathbb{E}) = 0$$

Задачи на собственные значения подразделяются на *частичную* (предполагает нахождение одного или нескольких λ из спектра) и *полную* (нахождение всего спектра собственных значений).

Рассматриваемые здесь численные методы решения задач на собственные значения подразделяются на две категории

1. Методы, направленные на получение коэффициентов характеристического полинома $det(\mathbb{A} - \lambda\mathbb{E})$ и на поиск его корней;

2. Методы, использующие преобразования подобия для приведения исходной матрицы $\mathbb{A}$ к треугольному или диагональному виду[13].

Методы первой категории предназначены для частичной проблемы собственных значений, а методы второй категории — для полной.

Этапы выполнения:

Ввод данных из текстового файла. Необходимо реализовать процедуру, считывающую из текстового файла данные о задаче[14].

Решение. Написать программу, реализующую заданный численный метод.

Проверка. Задав имя файла, содержащего данные о задаче, запустить программу и получить результат. Программа должна выводить на экран исходные данные, считанные из файла, промежуточные результаты[15], решение[16], номер итерации k, проверку[17].

Методы решения:

1. Метод Данилевского
2. Метод интерполяции
3. Метод $\mathbb{LU}$-разложения
4. Метод вращения Якоби
5. $\mathbb{QR}$-алгоритм
6. $\mathbb{QR}$-алгоритм на основе преобразования Хаусхолдера
7. $\mathbb{QR}$-алгоритм на основе вращений Гивенса

[13] Используются следующие свойства: (а) преобразование подобия сохраняет спектр; (б) собственные значения треугольных и диагональных матриц равны элементам, стоящим на главной диагонали.

[14] Подробнее см. аналогичный пункт раздела 5.1 «Прямые методы для задач линейной алгебры». Для задач на собственные значения вектор-столбец свободных членов $\vec{b}$ не задаётся; если же он присутствует в файле данных, то его следует игнорировать

[15] Для методов первой категории — график характеристического полинома; для методов второй категории — матрицы преобразований ($\mathbb{L}$, $\mathbb{U}$, $\mathbb{Q}$ и т.п.)

[16] Для методов первой категории — выбранную пару $\{\lambda_i^{(k)}, \vec{x}_i^{(k)}\}$; для методов второй категории — полный спектр $\vec{\lambda}^{(k)}$ и все соответствующие собственные вектора $\vec{x}_1^{(k)} \ldots \vec{x}_n^{(k)}$.

[17] Для каждой выведенной на экран пары $\{\lambda_i^{(k)}, \vec{x}_i^{(k)}\}$, вывести значение величины $det(\mathbb{A} - \lambda_i^{(k)}\mathbb{E})$ и невязку $\vec{\xi}_i^{(k)} = (\mathbb{A} - \lambda_i^{(k)}\mathbb{E})\vec{x}_i^{(k)}$.

6 Приближение функций

Задача приближения функций, в зависимости от практических целей, может быть сформулирована различным образом:

Приближение таблично заданной функции. Исходными данными являются значения функции на конечном наборе узлов. Требуется найти такую аналитическую формулу, которая наилучшим образом[18] описывала бы поведение приближаемой функции;

Приближение аналитически заданной функции. Подобные задачи возникают тогда, когда формула функции известна, но требует много ресурсов для вычисления. Необходимо найти более простую формулу, позволяющую вычислить значение исходной функции на некотором промежутке с достаточной точностью, но меньшими вычислительными затратами. Как правило, вычисляются значения исходной функции на некотором наборе узлов, и по этому набору строится приближение. То есть, задача сводится к предыдущему варианту — к приближению таблично заданной функции.

Обратная интерполяция. Если реализован алгоритм, который по таблице

x_i	x_0	x_1	$\ldots$	x_N
$f(x_i)$	f_0	f_1	$\ldots$	f_N

аппроксимирует функцию $f(x)$ на отрезке $[x_0, x_N]$, то, переставив местами строки в таблице и не меняя при этом алгоритм, получим аппроксимацию обратной функции $x(f)$. Знание обратной функции позволяет с лёгкостью получить решение нелинейного уравнения: для нахождения корня уравнения $f(x) = 0$ нужно всего лишь вычислить $x(0)$.

Вторая формулировка является более общей, более сложной, но и более удобной для отладки программ, реализующих приближение функций.

[18] В зависимости от практических целей, могут выдвигаться различные требования. Например, простота формулировки и вычисления (ИМЛ), аппроксимация производных (ИМЭ) и др.

6.1 Функции одной переменной

Во всех заданиях необходимо построить приближение по заданной аналитической функции[19] $f(x)$ методом, указанным преподавателем.

От того, какой метод будет задан, зависят постановка задачи и требования к оформлению. Ниже будут приведены варианты методов решения и более подробная формулировка заданий к ним.

В качестве первоначального отрезка интерполяции $[a,b]$ студент должен самостоятельно, по графику $f(x)$, выбрать любой отрезок, на котором $f(x)$ имеет 1 — 3 экстремума и не очень сильно меняется (если иное не указано в подробной формулировке задания).

Методы решения:

1. Интерполяция многочленом Лагранжа (ИМЛ)
2. Интерполяция многочленом Ньютона (ИМН)
3. Интерполяция многочленом Эрмита (ИМЭ)
4. Кубический сплайн
5. Обратная интерполяция (ИМН)
6. Обратная интерполяция (ИМЛ)
7. Наилучшее среднеквадратическое приближение

Интерполяция многочленами Лагранжа (ИМЛ) и Ньютона (ИМН)

Задание. Для указанных преподавателем функции и метода построить приближение на отрезке $[a,b]$ по узлам $x_0 \dots x_N$. Приближение необходимо строить по двум наборам узлов: равномерному и Чебышевскому.

Оформление. Параметры a, b и N должны определяться в разделе констант `const`. Программа должна строить три графика (в одних и тех же осях координат): график исходной функции, график многочлена, построенного на равномерном наборе узлов и график, соответствующий Чебышевскому набору. Все графики должны отображаться разными цветами, а узловые точки должны быть выделены дополнительно (например, кружочками).

[19] Варианты функций см. на стр.9.

Результат. Варьируя N и визуально анализируя получаемые графики, добиться оптимальной аппроксимации. Сделать выводы.

Интерполяция многочленом Эрмита (ИМЭ). Кубический сплайн

Задание. Для указанных преподавателем функции и метода построить кусочную интерполяцию на отрезке $[a,b]$ на равномерном наборе узлов $x_0 \ldots x_N$.

Оформление. Параметры a, b и N должны определяться в разделе констант `const`. Программа должна строить два графика (в одних и тех же осях координат): график исходной функции и график приближения. Графики должны отображаться разными цветами, а узловые точки должны быть выделены дополнительно (например, кружочками).

Результат. Варьируя N и визуально анализируя получаемые графики, добиться оптимальной аппроксимации. Сделать выводы.

Обратная интерполяция (ИМН, ИМЛ)

Задание. Для указанной преподавателем функции $f(x)$ построить график. По графику выбрать отрезок $[a,b]$, содержащий корень уравнения $f(x) = 0$. Задать N и, построив обратную интерполяцию на равномерном наборе узлов $x_0 \ldots x_N$, вычислить корень.

Оформление. Параметры a, b и N должны определяться в разделе констант `const`. Программа должна строить два графика (в одних и тех же осях координат): график исходной функции $f(x)$ и график приближения $x(f)$. Графики должны отображаться разными цветами, а узловые точки должны быть выделены дополнительно (например, кружочками).

Результат. Варьируя N и визуально анализируя получаемые графики, добиться оптимальной аппроксимации. Вывести на экран значение $x^* = x(0)$, вычисленное с помощью обратной интерполяции, и проверку $f(x^*)$. Сделать выводы.

Наилучшее среднеквадратическое приближение

Задание. Для указанной преподавателем функции $f(x)$ построить график. По равномерному набору узлов $x_0 \ldots x_N$ имитировать табличную функцию с погрешностями $\tilde{f}(\tilde{x})$ по следующему алгоритму[20]: $\tilde{x} := x + 0{,}001 - 0{,}002 * random$; $\tilde{f}(\tilde{x}) := f(\tilde{x}) + 0{,}01 - 0{,}02 * random$. Используя полученную таблицу $\tilde{f}(\tilde{x})$, построить приближение по методу наименьших квадратов многочленом $S(x)$ степени n.

Оформление. Параметры a, b, N и n должны определяться в разделе констант `const`. Программа должна строить два графика (в одних и тех же осях координат): график исходной функции $f(x)$ и график приближения $S(x)$. Графики должны отображаться разными цветами, а табличные значения $(\tilde{x}_i, \tilde{f}_i)$ должны быть изображены точками.

Результат. Варьируя степень приближающего многочлена n $(1 \leqslant n \leqslant 5)$ при достаточно большом N $(20 \leqslant N \leqslant 150)$ добиться наилучшего приближения. Сделать выводы.

6.2 Функции многих переменных

Этапы выполнения:

Задание. Для указанной преподавателем функции $f(x,y)$ построить двумерное приближение $P(x,y)$ в области $[a \leqslant x \leqslant b] \times [c \leqslant y \leqslant d]$. Область необходимо выбрать так, чтобы она содержала один локальный экстремум. Таблицу для интерполяции нужно заполнять с постоянным шагом по каждому из направлений (h_x и h_y) и с небольшим количеством шагов (3 — 5 шагов по каждому из направлений). Исключение составляет метод средних квадратов: для него, по аналогии с одномерной интерполяцией (см. предыдущий раздел), составить таблицу $\tilde{f}(\tilde{x},\tilde{y})$ с большим количеством узлов.

Оформление. Параметры a, b, c, d, N_x, N_y должны определяться в разделе констант `const`.

[20]Здесь «random» — это функция языка программирования `Pascal`, возвращает вещественное случайное число от 0 до 1.

Результат. Оценить точность приближения, вычислив

$$\sqrt{\sum_{i=0}^{10\cdot N_x}\sum_{j=0}^{10\cdot N_y}\left(f(a+i\cdot\hat{h}_x,c+j\cdot\hat{h}_y)-P(a+i\cdot\hat{h}_x,c+j\cdot\hat{h}_y)\right)^2}$$

где $\hat{h}_x = \frac{b-a}{10\cdot N_x}$, $\hat{h}_y = \frac{d-c}{10\cdot N_y}$.

Методы решения:

1. Треугольная интерполяция (многочлен Ньютона)
2. Последовательная интерполяция (многочлен Лагранжа)
3. Последовательная интерполяция (Сплайн)
4. Среднеквадратичное приближение

Варианты функций. Необходимо выполнить задание на интерполяцию для обеих функций ($f(x,y)$ и $g(x,y)$) из указанной преподавателем системы нелинейных уравнений (см. стр. 11).

7 Численное дифференцирование

Практические задачи на численное дифференцирование можно условно подразделить на две категории:

1. Аппроксимация производной на отрезке.
2. Аппроксимация производной в точке.

В первом случае результатом будет являться некоторая непрерывная функция, а во втором случае — одно-единственное значение в заданной точке или же полная таблица для всех узлов исходной дискретной величины.

Этапы выполнения:

Для методов 1-й категории

Задание. Для заданных метода и функции $f(x)$, построить графики: самой функции $f(x)$, аналитически вычисленной производной, а также все графики интерполяционных многочленов.

Оформление. Требования к оформлению графиков см. в разделе «Приближение функций», подпункт «Функции одной переменной»

Результат. Поварьировать расположение отрезка интерполяции $[a,b]$ и его длину. Сделать выводы.

Для методов 2-й категории

Задание. Для заданных метода и функции $f(x)$, построить графики: самой функции $f(x)$ и аналитически вычисленной производной. По формулам 2-го и 4-го порядков точности вычислить значение производной в точке $\tilde{x}$. Точность результата оценить и уточнить по формулам Рунге.

Оформление. Требования к оформлению графиков см. в разделе «Приближение функций», подпункт «Функции одной переменной». Значения малых величин (оценки погрешности) необходимо выводить на экран без каких-либо форматных ограничений[21]

Результат. Поварьировать расположение точки $\tilde{x}$ и длину шага между узлами разностной формулы h. Сделать выводы.

Методы:

1. Аппроксимация $f'(x)$ на основе ИМН степени 1, 2 и 3
2. Аппроксимация $f''(x)$ на основе ИМН степени 1 и 2
3. Аппроксимация $f''(x)$ на основе ИМЭ степени 3 и 5
4. Вычисление $f'(\tilde{x})$ в заданной точке $\tilde{x}$
5. Вычисление $f''(\tilde{x})$ в заданной точке $\tilde{x}$

Варианты функций см. на стр.9.

[21] Т.е., например, оставить величину в виде `9.56761818095963E-0002`, а не преобразовывать к виду `0.09568`

8 Численное интегрирование

«Численное интегрирование» — завершающая практическая работа семестрового курса «Вычислительная математика» и первого семестра годового курса «Численные методы». Выполнив предыдущие работы, студент должен был усвоить все основные требования к оформлению работы.

Не будем углубляться в подробности оформления и постараемся кратко сформулировать требования к выполнению данной работы:

1. Реализовать программно требуемый численный метод.
2. Проиллюстрировать в достаточной степени процесс решения.
3. Получить конечный результат.
4. Оценить, наглядно продемонстрировать точность полученного решения.
5. Сделать выводы.

Перечисленные требования оставляют довольно большую свободу выбора студенту. Однако, при написании и защите программы следует учесть, что от работы ожидается максимальная эффективность. Если по какому-либо из перечисленных пунктов работа выполнена недостаточно убедительно, то она будет возвращена на доработку с соответствующими замечаниями.

Методы решения:

1. Метод трапеций. Вычислить $\int\limits_a^b f(x)dx$, оценить эффективный порядок точности по Эйткену.
2. Метод Симпсона. Найти $\int\limits_a^b f(x)dx$ с заданной точностью.
3. Метод прямоугольников. Вычислить $\int\limits_a^b f(x)dx$, оценить эффективный порядок точности по Эйткену.
4. Метод Грегори. Найти $\int\limits_a^b f(x)dx$ и уточнить по формуле Рунге.
5. Сравнение методов трапеций и Симпсона. Реализовать программно оба метода, оценить точность по Рунге и сравнить.
6. Метод статистических испытаний, 1-й вариант.

7. Метод статистических испытаний, 2-й вариант.

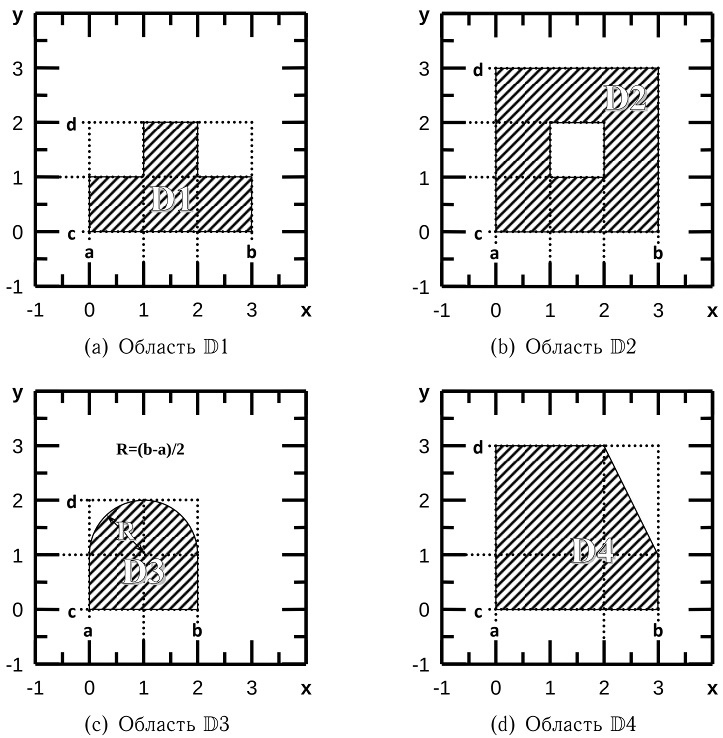
8. Метод ячеек. Вычислить $\int\limits_a^b \int\limits_c^d f(x,y)dxdy$, оценить порядок точности по Рунге.

9. Метод ячеек. Вычислить $\iint\limits_{\mathbb{D}} f(x,y)dxdy$ по области $\mathbb{D}$ сложной формы.

10. Последовательное интегрирование. Вычислить $\iint\limits_{\mathbb{D}} f(x,y)dxdy$ по области $\mathbb{D}$ сложной формы.

11. Метод статистических испытаний, 2-й вариант. Вычислить $\int\limits_a^b \int\limits_c^d f(x,y)dxdy$.

Варианты функций: для заданий на $\int f(x)dx$ см. стр.9; для заданий на $\iint f(x,y)dxdy$ см. функции, входящие в системы нелинейных уравнений на стр.11.



(a) Область $\mathbb{D}1$

(b) Область $\mathbb{D}2$

(c) Область $\mathbb{D}3$

(d) Область $\mathbb{D}4$

Рис. 1. Варианты сложных областей $\mathbb{D}$ для заданий на $\iint f(x,y)dxdy$.

9 Приложения

9.1 Правила оформления листинга программы

В документе, набранном с помощью Microsoft Word, листинг программы должен быть набран шрифтом Courier New размера 10, с выравниванием по левому краю.

```
Program Kurs; {Название программы. Необязательно}

Uses
  Crt,dos;

Const
  A=10;
  B=0.1;
  C:real=4;
  N=10;
{^ отступ на два символа}

Type {^ одна пустая строка между блоком
      Const(с комментариями) и блоком Type}

Vector=array[1..n] of real;

Var
  Matr1,Matr2:vector;

{Выше приведены описания используемых модулей, констант,
типов и переменных.  Ключевые слова  Uses,  Const, Type,
Var начинаются с первой позиции строки. Для всех позиций
в каждом блоке отступ на два символа.  Расстояние  между
блоками - одна пустая строка.}

Procedure FirstProcedure(a,b,c:real); {Комментарий.
                                       Необязательно}
Var
  Temp:real;
  I:integer;
Begin
  Temp:=random;
  For i:=1 to n-1 do
    Begin
      Matr1[i]:=temp;
```

```
      Matr2[i]:=temp*sin(temp)/cos(temp)*exp(temp)+
                ln(3-temp)+round(temp)+
                123456*cos (Pi/temp);
      {^^^^^^^^^^ Обратите внимание на то,
       как обрабатываются длинные строки!}
    End;
End; {Комментарий. Необязательно}

Procedure SecondProcedure; {Комментарий. Необязательно}
Begin
  C:=Pi;
End; {Комментарий. Необязательно}

{Выше приведено описание процедур. Функции оформляются
аналогично.  Внутри описания процедуры или функции не-
желательно оставлять пустые строки. Для каждого после-
дующего вложенного блока (begin...end,  repeat...until
и т.п.) делается отступ  на  два символа как  показано
выше. Между описаниями функций и процедур нужно остав-
лять одну пустую строку.  Комментарии  внутри описания
функций и процедур вставляйте по своему усмотрению, но
обязательно выравнивайте в соответствии с выравнивани-
ем текущего блока.}

begin{Начало тела программы}
  FirstProcedure(0.1,0.2,0.3);
  SecondProcedure;
end.{Конец тела программы}

{Тело программы, основной блок, оформляется в соответст-
вии  с  требованиями  к  оформлению  процедур и функций.
Begin...end (и все остальные ключевые слова, такие, как
uses, procedure, for и т.п.)  можно  (но необязательно)
выделить жирным шрифтом. Begin и end основного блока на-
чинаются с первой позиции строки.}
```

9.2 Нормы векторов и матриц

Нормы векторов

Нормой вектора $\vec{x} \in \mathbb{R}^n$ называют такое действительное число, обозначаемое $\|\vec{x}\|$, что:

1. $\|\vec{x}\| \geqslant 0$, причём $\|\vec{x}\| = 0 \Leftrightarrow \vec{x} = \vec{0}$;
2. $\|\lambda\vec{x}\| = |\lambda| \cdot \|\vec{x}\|\ \forall\lambda \in \mathbb{R}^1$;
3. $\|\vec{x} + \vec{y}\| \leqslant \|\vec{x}\| + \|\vec{y}\|\ \forall\vec{y} \in \mathbb{R}^n$.

$\|\vec{x}\|_p = \left(\sum\limits_{i=1}^{n} \|x_i\|^p\right)^{\frac{1}{p}} \qquad \forall p \in [1,\infty)$	l_p-норма, или норма Гёльдера.
$\|\vec{x}\|_2 = \sqrt{\sum\limits_{i=1}^{n} \|x_i\|^2}$	Евклидова норма. Получается из нормы Гёльдера l_p при $p = 2$. Другое название — сферическая норма (по названию поверхности, определяемой уравнением $\|\vec{x}\| = \text{const}$).
$\|\vec{x}\|_1 = \sum\limits_{i=1}^{n} \|x_i\|$	Норма-сумма, или октаэдрическая норма. Получается из l_p при $p = 1$.
$\|\vec{x}\|_{=\|\vec{x}\|_\infty = \max\limits_{i=\overline{1,n}} \|x_i\|}$	Норма-максимум, или кубическая норма. Получается из l_p при $p \to \infty$.

Таблица 1. Общеупотребительные нормы векторов

Между разными нормами существует соотношение

$$\|\vec{x}\|_c \leqslant \|\vec{x}\|_2 \leqslant \|\vec{x}\|_1 \leqslant n\,\|\vec{x}\|_c$$

Иногда нормы l_p и C определяют по-другому:

$$\|\vec{x}\|_p = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^p\right)^{\frac{1}{p}}, \qquad \|\vec{x}\|_c = \lim_{p\to\infty} \|\vec{x}\|_p\,,$$

тогда соотношение между нормами будет[22]

$$\|\vec{x}\|_1 \leqslant \|\vec{x}\|_2 \leqslant \|\vec{x}\|_c \leqslant \sqrt{n}\,\|\vec{x}\|_2 \leqslant n\,\|\vec{x}\|_1$$

[22] переопределив норму l_p, мы, тем самым, переопределили и l_2, которая есть l_p при $p = 2$

Нормы матриц

Нормой матрицы $\mathbb{A}$ называют такое действительное число, обозначаемое $\|\mathbb{A}\|$, что:

1. $\|\mathbb{A}\| \geqslant 0$, причём $\|\mathbb{A}\| = 0 \Leftrightarrow \mathbb{A} = 0$;
2. $\|\lambda\mathbb{A}\| = |\lambda| \cdot \|\mathbb{A}\| \; \forall\lambda \in \mathbb{R}^1$;
3. $\|\mathbb{A} + \mathbb{B}\| \leqslant \|\mathbb{A}\| + \|\mathbb{B}\|$;
4. $\|\mathbb{A}\mathbb{B}\| \leqslant \|\mathbb{A}\| \cdot \|\mathbb{B}\|$.

($\mathbb{B}$ — произвольная $n \times n$-матрица).

Иногда норму матрицы определяют только с помощью трёх первых условий; в таком случае говорят, что вышеприведённое определение задаёт ***мультипликативную*** норму.

Норма матрицы называется ***согласованной*** с нормой вектора, если $\forall\vec{x}$:

$$\|\mathbb{A}\vec{x}\| \leqslant \|\mathbb{A}\| \cdot \|\vec{x}\|$$

Норма, удовлетворяющая условию

$$\|\mathbb{A}\| = \max_{\vec{x}\neq 0} \frac{\|\mathbb{A}\vec{x}\|}{\|\vec{x}\|}$$

(т.е., — наименьшая согласованная) называется ***подчинённой*** норме вектора.

$\|\mathbb{A}\|_1 = \max\limits_j \sum\limits_{i=1}^{n} \left\|a_{ij}\right\|$	Без специального названия. Подчинённая для $\|\vec{x}\|_1$
$\|\mathbb{A}\|_2 = \sqrt{\max\limits_i \|\lambda_i\|}$	Спектральная норма. (Здесь λ_i — собственные значения). Подчинена норме $\|\vec{x}\|_2$
$\|\mathbb{A}\|_E = \sqrt{\sum\limits_{i,j=1}^{n} \left\|a_{ij}\right\|^2}$	Евклидова, или сферическая. Согласованная с $\|\vec{x}\|_2$, но не подчинённая ей.
$\|\mathbb{A}\|_c = \max\limits_i \sum\limits_{j=1}^{n} \left\|a_{ij}\right\|$	Без специального названия. Подчинённая для $\|\vec{x}\|_c$
$\|\mathbb{A}\|_M = n \max\limits_{i,j} \left\|a_{ij}\right\|$	Максимальная норма. Согласована со всеми рассмотренными нормами векторов.

Таблица 2. Общеупотребительные нормы матриц

Список литературы

1. *Демидович Б. П., Марон И. А., Шувалова Э. З.* Численные методы анализа. Приближение функций, дифференциальные и интегральные уравнения: Учеб.пособие для студ.втузов / Под ред. Б. П. Демидовича. — 3-е, перераб. изд. — М.: Наука, 1967. — 144 с.

2. *Бахвалов Н. С.* Численные методы. Анализ, алгебра, обыкновенные дифференциальные уравнения : Учебное пособие. — 2-е, стереотип. изд. — М.: Наука, 1975. — 632 с.

3. *Мудров А. Е.* Численные методы для ПЭВМ на языках Бейсик, Фортран и Паскаль. — Томск: МП «Раско», 1991. — 272 с.

4. *Заварыкин В. М., Житомирский В. Г., Лапчик М. П.* Численные методы : Учеб.пособие для студентов физ-мат спец. — М.: «Просвещение», 1991. — 176 с.

5. *Демидович Б. П., Марон И. А., Шувалова Э. З.* Численные методы анализа.Приближение функций,дифференциальные и интегральные уравнения : Учебное пособие для ВТУзов / Под ред. Б. П. Демидовича. — 2-е, испр. и доп. изд. — М.: Физматгиз, 1963. — 400 с.

6. *Пирумов У. Г.* Численные методы : Учебное пособие для студентов высших учебных заведений. — 3-е, испр. и доп. изд. — М.: Дрофа, 2003. — 221 с.

7. *Волков Е. А.* Численные методы : Учебное пособие. — СПб.: Лань, 2004. — 248 с.

8. *Бахвалов Н. С., Жидков Н. П., Кобельков Г. М.* Численные методы [[Текст]]. — 3-е, доп. и перераб. изд. — М.: Бином. Лаборатория знаний, 2004. — 636 с.

9. *Вержбицкий В. М.* Численные методы [[Текст]] : Линейная алгебра и нелинейные уравнения : учебное пособие для студентов высших учебных заведений. — М.: Высшая школа, 2000. — 265 с.

10. *Лапчик М. П., Рагулина М. И., Хеннер Е. К.* Численные методы [[Текст]] : учебное пособие для студентов высших учебных заведений / Под ред. М. П. Лапчика. — М.: Academia, 2004. — 383 с.

Д. А. Моргун, А. Г. Назин

Вычислительная математика

Методические указания по выполнению практических работ для студентов факультета информационных технологий специальностей 010200(010501) «Прикладная математика и информатика» и 654600(230102)«Автоматизированные системы обработки информации и управления»

Часть 2

Оригинал-макет подготовлен авторами

Подписано в печать 20.12.2006. Формат 60 × 84/16.
Усл. печ. л. 1,86. Уч.-изд. л. 1,49
Печать трафаретная. Тираж 60.

Отпечатано полиграфическим отделом ООО «Авиаграфия»
г. Сургут, ул. Профсоюзов, 37. Тел. (3462) 32-33-32.