

Вычислительная математика

Осень 2019

Лабораторная работа 1, прислать до 29.09.2018 23:59

Преподаватель: Маловичко М. С.

1. LU разложение

- Написать функцию, выполняющую LU разложение вещественной квадратной матрицы без выбора главного элемента произвольного размера.
- Написать программу, решающую СЛАУ $Ax = f$ с помощью LU разложения.
- Вычислить решение x . Вычислить невязку $e = \|Ax - f\|/\|f\|$ и убедиться, что она мала ($e < 10^{-6}$).

При выполнении задания:

- Язык: Python/C/C++.
- Входные данные взять из прилагаемых файлов: матрица 3×3 и правая часть в файлах An.txt и fn.txt, где n – номер варианта.
- Полагать, что элементы матриц в памяти упорядочены по столбцам и матрицы в файлах записаны по столбцам;
- Не использовать сторонние библиотеки для реализации численного метода (numpy, boost, BLAS/LAPACK и т.п.)
- Программа должна состоять из одного исходного файла.

Выслать мне на почту:

- Отчёт в PDF¹, который должен обязательно содержать следующее: выписанные матрицы A, L, U , правая часть f , решение x , невязка e .
- Исходный файл с программой *.py/*.c/*.cpp, из которого можно собрать программу (1 исходник!).
- Входные данные (A, f), промежуточные данные (L, U), результат (x). При записи в файл вывод не форматировать! Пример вывода матрицы 3×3 :

-----L.txt-----

```
1.000000e+00 <--это первый столбец
6.666667e-02
6.666667e-02
0.000000e+00 <--это второй столбец
1.000000e+00
5.511811e-02
0.000000e+00 <--это третий столбец
0.000000e+00
1.000000e+00
```

¹ Создать PDF можно любым способом, например, сделать документ MsWord и сохранить как PDF.

2. Прогонка для трёхдиагональных систем

- Написать функцию, решающую методом прогонки трёхдиагональную СЛАУ произвольного размера n .
- Написать программу, решающую СЛАУ $Ax = c$ с помощью этой функции.
- Вычислить решение x . Вычислить невязку $e = \|Ax - f\|/\|f\|$ и убедиться, что она мала ($e < 10^{-6}$).
- Вычислить решение x_G методом Гаусса из предыдущей части, и убедиться, что решения совпадают.

При выполнении задания:

- Язык Python/C/C++
- Запрещено аллоцировать матрицу $n \times n$!
- Входные данные взять из прилагаемых файлов: трёхдиагональная матрица 1000×1000 в виде трёх диагоналей an_diag.txt (нижняя), cn_diag.txt(главная), bn_diag.txt(верхняя) и правая часть rn.txt, где n – номер варианта.
- Не использовать сторонние библиотеки (numpy, boost, BLAS/LAPACK и т.п.)
- Программа должна состоять из одного исходного файла *.py/*.c/*.cpp

Выслать мне на почту:

- Отчёт в PDF², который должен обязательно содержать следующее. На одном графике: правая часть f (в логарифмическом масштабе по Y). На другом графике: решения x и x_G (в логарифмическом масштабе). Невязка e .
- Исходный файл с кодом *.py/*.c/*.cpp, из которого можно собрать программу (один !).
- Прислать файлы: входные данные (диагонали a_diag, c_diag, b_diag и правая часть r), промежуточные данные (массивы прогоночных коэффициентов alpha и beta), результат (x).

² Создать PDF можно любым способом, например, сделать документ MsWord и сохранить как PDF.