

Контакты:

Лектор - ivisaev@corp.ifmo.ru

P3202, P3210, P3211, P3218 - maria.petrova@corp.ifmo.ru

P3200, P3201, P3217, P3219, P3275 - ovkalyonova@corp.ifmo.ru

Лабораторные сдаются преподавателю своей группы по расписанию группы.

!! Варианты можно получить только лично - на паре. В случае, если вам не терпится приступить к выполнению лабораторной работы, а варианта у вас нет - можете сделать все варианты =)

Литература:

- Демидович Б.П. Марон И.А. Основы вычислительной математики
- Л.И. Турчак "Основы численных методов"
- Дж. Форсайт, М. Малькольм, К. Моултер "Машинные методы математических вычислений"
- [Требования к блок-схемам. ЕСПД. Схемы алгоритмов, программ, данных и систем.](#)

Семестр:

- 5 Лабораторных + РК

Сроки сдачи (без снижения баллов): - 2 занятия на лабораторную

//Группа нечетная неделя - нечет

//Группа четная неделя - чет

- Лаб1 - нечет ~ 11 октября
чет ~ 4 октября
- Лаб2 - нечет ~ 8 ноября
чет ~ 1 ноября
- Лаб3 - нечет ~ 6 декабря
чет ~ 29 ноября
- Лаб4 ~ 27 декабря
- Лаб 5 - январь, доп. пара

Для получения зачёта необходимо сдать все лабораторные + написать РК.

!!!!В случае если 4 лабораторные сданы до следующей даты включительно + Сдан тест(РК) с первого раза Лаб5 делать не нужно - оценка за Лаб5 ставится по согласованию с преподавателем (по лаб.раб.) :

- ☐ нечет - 6 декабря
- ☐ чет - 29 ноября

Общее:

- С#, Java, C++, иное обговаривается лично с преподавателем
- Конечному пользователю должна быть представлена интуитивно понятная программа, без значительных дефектов (плавающие поля, бесконечно множасьи ячейки, обработанный ввод данных и т.д.).
- В лабораторных, предполагающих отображение графиков - график должен полностью отображать весь заданный интервал (с запасом)

В Программе численный метод должен быть в виде отдельной подпрограммы или класса, в который исходные данные передаются в качестве параметров, выходные - тоже (либо возвращаемое значение).

Ввода -вывода в классе(подпрограмме), где реализован сам численный

метод, быть не должно(учимся писать код так, чтобы можно было повторно использовать).

В отчете должно быть:

- описание метода,расчетные формулы, прямое и обратное действие
- листинг программы(по крайней мере где реализован сам класс)

- блок-схема численного метода(можно и всей программы) [см. требования к оформлению блок-схем]
- примеры и результаты работы программы
- **выводы**

Лабораторная работа 1 (Решение системы линейных алгебраических уравнений СЛАУ)

Варианты:

- Метод Гаусса
- Метод Гаусса с выбором главного элемента
- Метод простых итераций
- Метод Гаусса-Зейделя

Размерность $n \leq 20$ (задается из файла или с клавиатуры - по выбору конечного **пользователя**)

Должно быть предусмотрено чтение исходных данных как из файла, так и ввод с клавиатуры.

Должна быть реализована возможность ввода коэффициентов матрицы как с клавиатуры, так и из файла.

Также предусмотреть случайные коэффициенты.

Обязательно: Тестовые данные на матрице большого размера (5×5 / 6×6 ...) + в отчёт с решением.

Для точных методов(Гаусс и главные элементы) должно быть реализовано:

- Вычисление определителя
- Вывод треугольной матрицы (включая преобразованный столбец В)
- Столбец неизвестных
- Столбец невязок

Для итерационных методов:

- Точность задается с клавиатуры/файла
- Проверка диагонального преобладания
//В случае, если диагональное преобладание в изначальной матрице отсутствует - предлагается сделать перестановку строк/столбцов до тех пор, пока преобладание не будет достигнуто. В случае невозможности достижения диагонального преобладания - выводить сообщение.
- Столбец неизвестных
- Количество итераций, за которое было найдено решение
- Столбец погрешностей

Лабораторная работа 2 (Интегрирование)

Варианты:

- Метод прямоугольников
(должен быть реализован расчет 3мя модификациями: левые, правые, средние)
- Метод трапеций
- Метод Симпсона

Пользователь выбирает функцию, интеграл которой он хочет вычислить (3-5 функций), из тех, которые предлагает программа.

В численный метод должен быть передан параметр-агрегат на подпрограмму вычисления значения функции в точке x .

Пользователь задает пределы интегрирования и точность.

NOTE! Если нижний предел интегрирования \geq верхнего предела - интеграл должен считаться корректно!

В результате должны получить:

- значение интеграла
- количество разбиений, на которое пришлось разбить
- полученную погрешность

Для оценки погрешности использовать оценку Рунге.

Лабораторная работа 3 (Приближение функций)

Варианты:

- Интерполирование многочленом Лагранжа
- Интерполирование многочленом Ньютона
- Интерполирование кубическими сплайнами
- Аппроксимация методом наименьших квадратов

Для **интерполяции** необходимо подготовить 3-4 набора данных (в зависимости от функции).

/*Исходные данные должны быть подготовлены следующим образом:

- Берем функцию
- Берем точки x (точки не обязательно упорядочены)
- значение y получаем на основе данных выбранной функции

Например:

- берем $\sin x$
- 1) берем 3-4 точки на интервале 0 по 2π (шаг более менее большой)
- 2) берем 8-10 точек на интервале 0 по 2π (уменьшаем шаг)
- 3) точки с предыдущего примера, только для одной точки изменяем значение y , например было 0.8, делаем -5, смотрим как ведет себя интерполяция.
- 4) берем 8-10 точек на интервале 0 по 50π

*/

В итоге, должны получить график, на котором одним цветом исходная функция ($\sin x$), а другим цветом полученный график в результате интерполяции, и на графике должны быть отмечены сами **точки** (узлы) интерполяции.

Интерполяционный график должен пройти через исходные эти точки.

Программа должна позволять найти значение y (отдельное поле) для любого введенного x (рассчитывается на основе построенного интерполяционного многочлена).

Для **аппроксимации**:

По пунктам так же как для интерполяции можно написать

1. Задается произвольный набор значений пар (x, y)
2. Задается аппроксимирующая функция
3. Рассчитываются программой коэффициенты аппроксимирующей функции
4. Производится вычисление точки с наибольшим отклонением
5. Найденная точка исключается. Производится перерасчет коэффициентов аппроксимирующего многочлена (см. п.3).
6. Строится график, содержащий в себе две функции (1 - до исключения, 2 - после исключения и пересчёта) и набор заданных изначально точек (пар значений (x, y))

7. Помимо этого, отдельно на экран выводятся полученные значения аппроксимирующих коэффициентов
Рассчитанные два раза коэффициенты аппроксимирующей функции также должны быть выведены на экран.

Лабораторная работа 4 (Решение ОДУ(задача Коши))

Варианты:

- Метод Эйлера
- Усовершенствованный метод Эйлера
- Метод Рунге-Кутты 4 го порядка

многошаговые методы:

- Адамса
- Предиктора и Корректора
- Милна.

Задается ОбДифУр вида $y' + f(x,y) = 0$, пользователь задает начальные условия (x_0, y_0) , конец отрезка и точность.

Программа сама вычисляет шаг в зависимости от точности для нахождения массива значений x и y .

Используя интерполирование 3-й работы строим график.

У кого 3-я работа была аппроксимация, строит график по полученным данным, задав очень маленькую точность.

//Не забудьте вставить примеры в отчет!

+ пару примеров ОДУ - с заранее известным решением вида $y = f(x)$, где изначально y' зависело не только от x .

Лабораторная работа 5 доступна по [ссылке](#)