

ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В.Ф. УТКИНА»

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ «Специальные главы математики»

Направление подготовки

09.04.02 Информационные системы и технологии

ОПОП академической магистратуры

«Информационные системы и технологии»

Квалификация (степень) выпускника – магистр

Формы обучения – очная, очно-заочная

Рязань

1. ПЛАНЫ ЛАБОРАТОРНЫХ И ПРАКТИЧЕСКИХ РАБОТ

Практическая работа № 1

Линейная аппроксимация

Цель работы: изучение метода линейной аппроксимации и его реализация на языке программирования C#

Задание

Пусть при изучении неизвестной функциональной зависимости между x и y , полученной в результате серии экспериментов, была сформирована таблица значений (табл. 1). Необходимо найти приближенную функциональную зависимость методом линейной аппроксимации и определить значения параметров аппроксимирующей функции.

Таблица 1 – Экспериментальная зависимость

x_i	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0	10,0
y_i	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8

Выполнение работы

Шаг 1. Расчет коэффициентов «вручную». При выборе в качестве аппроксимирующей функции многочлена первой степени необходимо построить линейную модель $f = a_1x + a_0$, которая наилучшим образом будет описывать наблюдаемые значения. Далее, используя метод наименьших квадратов, найдем значения коэффициентов аппроксимирующей функции: a_1 и a_0 . Для этого вычислим:

$$s_1 = \sum_{i=1}^{10} x_i = 1,0 + 2,0 + 3,0 + 4,0 + 5,0 + 6,0 + 7,0 + 8,0 + 9,0 + 10,0 = 55,0,$$

$$s_2 = \sum_{i=1}^{10} x_i^2 = 1,0^2 + 2,0^2 + 3,0^2 + 4,0^2 + 5,0^2 + 6,0^2 + 7,0^2 + 8,0^2 + 9,0^2 + 10,0^2 = 385,0,$$

$$s_3 = \sum_{i=1}^{10} y_i = 7,4 + 9,5 + 11,1 + 12,9 + 14,6 + 17,3 + 18,2 + 20,7 + 22,5 + 23,8 = 158,0,$$

$$s_4 = \sum_{i=1}^{10} x_i y_i = 1,0 \cdot 7,4 + 2,0 \cdot 9,5 + 3,0 \cdot 11,1 + 4,0 \cdot 12,9 + 5,0 \cdot 14,6 + 6,0 \cdot 17,3 + 7,0 \cdot 18,2 + 8,0 \cdot 20,7 + 9,0 \cdot 22,5 + 10,0 \cdot 23,8 = 1021,6.$$

По формулам (1.6) – (1.8) вычислим определители:

$$\Delta = s_2 \cdot 10 - s_1^2 = 385,0 \cdot 10 - 55,0^2 = 825,0,$$

$$\Delta_{a_0} = s_2 s_3 - s_1 s_4 = 385,0 \cdot 158,0 - 55,0 \cdot 1021,6 = 4642,0,$$

$$\Delta_{a_1} = s_4 \cdot 10 - s_3 s_1 = 1021,6 \cdot 10 - 158,0 \cdot 55,0 = 10216 - 8690 = 1526,0.$$

Затем находим значения параметров:

$$a_0 = \frac{\Delta_{a_0}}{\Delta} = \frac{4642,0}{825,0} = 5,626666667,$$

$$a_1 = \frac{\Delta_{a_1}}{\Delta} = \frac{1526,0}{825} = 1,84969697.$$

С учетом полученных значений параметров аппроксимирующая функция приняла вид:

$$f = 1,84969697 \cdot x + 5,626666667.$$

Оценим точность линейной аппроксимации для рассмотренной зависимости. В табл. 2 приведены результаты вычислений.

Таблица 2

Результаты вычислений для линейной аппроксимации

№	1	2	3	4	5	6	7	8	9	10
x_i	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0	10,0
y_i	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8
$f_i = a_1 x_i + a_0$	7,4764	9,3261	11,176	13,026	14,875	16,725	18,575	20,424	22,274	24,124
$\varepsilon_i = f_i - y_i$	0,0764	-0,1739	0,0758	0,1255	0,2752	-0,5752	0,3745	-0,2758	-0,2261	0,3236

Определим меру отклонения E :

$$E = \sum_{i=1}^{10} \varepsilon_i^2 = 0,8363182.$$

Как видно из табл. 2, максимальная погрешность $|\Delta_{\max}| = 0,5752$.

Шаг 2. Построение блок-схемы алгоритма.

Шаг 3. Реализация линейной аппроксимации на языке C#. Инициализация таблицы значений описывается следующим фрагментом программы.

```
public Form1()
{
    InitializeComponent();
    dataGridView1.ColumnCount = 11;
    dataGridView1.RowCount = 6;
    for (int i=1; i<=10; i++)
        dataGridView1.Rows[0].Cells[i].Value = i;
    dataGridView1.Rows[1].Cells[0].Value = "x";
    dataGridView1.Rows[2].Cells[0].Value = "y";
    dataGridView1.Rows[3].Cells[0].Value = "f=a1*x+a0";
    dataGridView1.Rows[4].Cells[0].Value = "eps";
    // Задание значений функциональной зависимости
    // Значения xi
    dataGridView1.Rows[1].Cells[1].Value = 1.0;
    dataGridView1.Rows[1].Cells[2].Value = 2.0;
    dataGridView1.Rows[1].Cells[3].Value = 3.0;
    dataGridView1.Rows[1].Cells[4].Value = 4.0;
    dataGridView1.Rows[1].Cells[5].Value = 5.0;
    dataGridView1.Rows[1].Cells[6].Value = 6.0;
    dataGridView1.Rows[1].Cells[7].Value = 7.0;
    dataGridView1.Rows[1].Cells[8].Value = 8.0;
    dataGridView1.Rows[1].Cells[9].Value = 9.0;
    dataGridView1.Rows[1].Cells[10].Value = 10.0;
    // Значения yi
    dataGridView1.Rows[2].Cells[1].Value = 7.4;
    dataGridView1.Rows[2].Cells[2].Value = 9.5;
    dataGridView1.Rows[2].Cells[3].Value = 11.1;
    dataGridView1.Rows[2].Cells[4].Value = 12.9;
```

```

dataGridView1.Rows[2].Cells[5].Value = 14.6;
dataGridView1.Rows[2].Cells[6].Value = 17.3;
dataGridView1.Rows[2].Cells[7].Value = 18.2;
dataGridView1.Rows[2].Cells[8].Value = 20.7;
dataGridView1.Rows[2].Cells[9].Value = 22.5;
dataGridView1.Rows[2].Cells[10].Value = 23.8;
}

```

Блок объявления переменных принял вид:

```

// объявление переменных
double[] x = new double [10]; // массив значений  $x_i$ 
double[] y = new double [10]; // массив значений  $y_i$ 
double s1,s2,s3,s4; // суммы
double[] f = new double [10]; // массив аппроксимированных значений  $f_i$ 
double[] eps = new double[10]; // массив погрешностей  $e_i$ 
double E; // сумма квадратов погрешностей
double etax; // максимальная абсолютная погрешность
double d; // главный определитель
double da0,da1; // определители
double a0,a1; // параметры аппроксимации

```

Функция *opred2()*, вычисляющая определитель 2×2 , имеет следующее описание.

```

private double opred2(double a11, double a12, double a21, double a22)
{
    return a11 * a22 - a12 * a21;
}

```

Считывание массивов данных с формы приложения описывается следующим кодом.

```

// считывание массивов с формы приложения
for (int i = 0; i < 10; i++)
{
    x[i] = Convert.ToDouble(dataGridView1.Rows[1].Cells[i+1].Value);
    y[i] = Convert.ToDouble(dataGridView1.Rows[2].Cells[i + 1].Value);
}

```

Вычисление сумм описывает следующий фрагмент.

```

// вычисление сумм
s1 = 0.0; // сумма  $x_i$ 
s2 = 0.0; // сумма квадратов  $x_i$ 
s3 = 0.0; // сумма  $y_i$ 
s4 = 0.0; // сумма  $x_i \cdot y_i$ 
for (int i = 0; i < 10; i++)
{
    s1 += x[i];
    s2 += x[i] * x[i];
    s3 += y[i];
    s4 += x[i] * y[i];
}

```

Вычисление определителей приняло вид:

```
// вычисление определителей
d = opred2(s2,s1,s1,10);
da0 = opred2(s2, s4, s1, s3);
da1 = opred2(s4, s1, s3, 10);
```

Вычисление параметров a_0 и a_1 описывается следующим фрагментом.

```
// вычисление параметров a0 и a1
a0 = da0 / d;
a1 = da1 / d;
```

Расчет аппроксимирующих значений и вычисление погрешности осуществляются следующим образом.

```
// расчет аппроксимирующих значений и вычисление погрешности
E = 0.0;
emax = 0.0;
for (int i = 0; i < 10; i++)
{
    f[i] = a1 * x[i] + a0;
    eps[i] = f[i] - y[i];
    E += eps[i] * eps[i];
    if (Math.Abs(eps[i]) > emax)
        emax = Math.Abs(eps[i]);
}
}
```

Вывод результатов в таблицу выполняется следующим текстом программы.

```
// вывод результатов в таблицу
for (int i = 0; i < 10; i++)
{
    dataGridView1.Rows[3].Cells[i + 1].Value = f[i];
    dataGridView1.Rows[4].Cells[i + 1].Value = eps[i];
}
textBox1.Text += E;
textBox2.Text += emax;
textBox3.Text += a0;
textBox4.Text += a1;
```

Результат работы программы по расчету линейной аппроксимации показан на рис. 2.

	1	2	3	4	5	6	7	8	9	10
x	1	2	3	4	5	6	7	8	9	10
y	7.4	9.5	11.1	12.9	14.6	17.3	18.2	20.7	22.5	23.8
f=a1*x+a0	7.47636363636364	9.32606060606061	11.1757575757576	13.0254545454545	14.8751515151515	16.7248484848485	18.5745454545455	20.4242424242424	22.2739393939394	24.1236363636364
eps	0.07636363636362	-0.173939393939394	0.075757575757576	0.125454545454545	0.275151515151515	-0.5751515151514	0.374545454545455	-0.275757575757577	-0.226060606060607	0.323636363636364

button1

E= 0,836242424242

emax= 0,575151515151

a0= 5,62666666666667

a1= 1,84969696969697

Рис. 2. Результат работы программы линейной аппроксимации

Как видно из рис. 2, полученные результаты совпадают с расчетными значениями. Следовательно, программа работает верно.

Контрольные вопросы

1. Задачи аппроксимации.
2. Способы аппроксимации.
3. Метод наименьших квадратов.
4. Расчет параметров линейной аппроксимации.
5. Расчет параметров квадратичной аппроксимации.
6. Блок-схема алгоритма расчета линейной аппроксимации.
7. Блок-схема алгоритма расчета квадратичной аппроксимации.
8. Вычисление линейной аппроксимации на языке C#.
9. Вычисление квадратичной аппроксимации на языке C#.

Варианты заданий

№ вар.	1	2	3	4	5	6	7	8	9	10	11	12	13
x_i	y_i												
1,0	7,1	7,2	7,3	7,5	7,6	7,7	7,8	7,9	8,0	8,1	8,2	8,3	8,4
2,0	8,6	8,7	8,9	9,0	9,2	9,3	9,4	9,6	9,7	9,9	10,0	10,2	10,3
3,0	10,4	10,7	11,0	11,2	11,4	11,6	11,8	12,0	12,1	12,2	12,3	12,4	12,5
4,0	12,6	12,7	12,8	13,0	13,2	13,4	13,6	13,9	14,2	14,5	14,8	15,1	15,4
5,0	15,5	15,6	15,7	15,8	15,9	16,0	16,1	16,2	16,3	16,4	16,5	16,6	16,7
6,0	16,8	17,0	17,2	17,4	17,6	17,8	18,0	18,2	18,4	18,6	18,8	19,0	19,2
7,0	19,3	19,5	19,4	19,6	19,7	19,9	19,8	20,0	20,1	20,3	20,2	20,4	20,5
8,0	20,6	20,8	21,0	21,2	21,4	21,6	21,8	22,0	22,2	22,4	22,6	22,8	23,0
9,0	21,6	21,9	22,0	22,2	22,5	22,7	22,9	23,1	23,3	23,5	23,7	23,9	24,1
10,0	23,0	23,2	23,4	23,6	23,9	24,1	24,3	24,5	24,7	24,9	25,1	25,3	25,5

Содержание отчета

1. Название лабораторной работы.
2. Цель лабораторной работы.
3. Теоретические сведения.
4. Расчет линейной аппроксимации для заданной функции «вручную».
5. Блок-схема алгоритма вычисления линейной аппроксимации.
6. Текст подпрограммы линейной аппроксимации.
7. Результаты работы подпрограмм линейной аппроксимации.
8. Анализ полученных результатов.

Библиографический список

1. Челебаев С.В. Аппроксимация функциональных зависимостей двух переменных: метод. указ. к практ. работе: Методические указания. Рязань: РИЦ РГРТУ, 2020. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2636>.
2. Челебаев С.В. Методы аппроксимации функций: Методические указания. Рязань: РИЦ РГРТУ, 2019. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2312>.

Практическая работа № 2 Квадратичная аппроксимация

Цель работы: изучение метода квадратичной аппроксимации и его реализация на языке программирования C#.

Задание

Пусть при изучении неизвестной функциональной зависимости между x и y , полученной в результате серии экспериментов, была сформирована таблица значений. Необходимо найти приближенную функциональную зависимость методом квадратичной аппроксимации и определить значения параметров аппроксимирующей функции.

x_i	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0	10,0
y_i	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8

Решение

Рассмотрим изменения и добавления по сравнению с программой для линейной аппроксимации.

1. Изменение количества строк в таблице с целью добавления результатов квадратичной аппроксимации.

```
dataGridView1.RowCount = 8; // изменено поле
```

2. Добавление подписей к новым строкам таблицы.

```
dataGridView1.Rows[5].Cells[0].Value = "f2=a2*x*x+a1*x+a0";  
dataGridView1.Rows[6].Cells[0].Value = "eps2";
```

3. Определение функции по расчету определителя 3*3.

```
private double opred3(double a11, double a12, double a13, double a21, double a22, double a23,  
double a31, double a32, double a33)  
{  
    return a11 * opred2(a22,a23,a32,a33) - a21 * opred2(a12,a13,a32,a33) + a31 *  
    opred2(a12,a13,a22,a23);  
}
```

4. Объявление дополнительных переменных.

```
double s5, s6, s7; // суммы  
double da2; // определитель  
double a2; // параметр аппроксимации
```

5. Переопределение блока вычисления сумм.

```
s1 = 0.0; // сумма xi  
s2 = 0.0; // сумма xi*xi  
s3 = 0.0; // сумма xi*xi*xi  
s4 = 0.0; // сумма xi*xi*xi*xi  
s5 = 0.0; // сумма yi  
s6 = 0.0; // сумма xi*yi  
s7 = 0.0; // сумма xi*xi*yi  
for (int i = 0; i < 10; i++)  
{  
    s1 += x[i];  
    s2 += x[i] * x[i];  
    s3 += x[i] * x[i] * x[i];  
    s4 += x[i] * x[i] * x[i] * x[i];
```

```

s5 += y[i];
s6 += x[i] * y[i];
s7 += x[i] * x[i] * y[i];
}

```

6. Переопределение блока вычисления определителей.

```

d = opred3(10,s1,s2,s1,s2,s3,s2,s3,s4);
da0 = opred3(s5,s1,s2,s6,s2,s3,s7,s3,s4);
da1 = opred3(10,s5,s2,s1,s6,s3,s2,s7,s4);
da2 = opred3(10,s1,s5,s1,s2,s6,s2,s3,s7);

```

7. Добавление блока вычисления параметра a2.

```

a2 = da2 / d;

```

8. Переопределение блока расчета аппроксимирующих значений и вычисления погрешности.

```

E = 0.0;
emax = 0.0;
for (int i = 0; i < 10; i++)
{
    f[i] = a2 * x[i] * x[i] + a1 * x[i] + a0;
    eps[i] = f[i] - y[i];
    E += eps[i] * eps[i];
    if (Math.Abs(eps[i]) > emax)
        emax = Math.Abs(eps[i]);
}

```

9. Блок вывода результата в таблицу.

```

for (int i = 0; i < 10; i++)
{
    dataGridView1.Rows[5].Cells[i + 1].Value = f[i];
    dataGridView1.Rows[6].Cells[i + 1].Value = eps[i];
}
textBox5.Text += E;
textBox6.Text += emax;
textBox7.Text += a0;
textBox8.Text += a1;
textBox9.Text += a2;

```

Результат работы программы по расчету квадратичной аппроксимации показан на рис. 3.

	1	2	3	4	5	6	7	8	9	10
x	1	2	3	4	5	6	7	8	9	10
y	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8
f=a1*x+a0	7,47636363636364	9,32606060606061	11,1757575757576	13,0254545454545	14,8751515151515	16,7248484848485	18,5745454545455	20,4242424242424	22,2739393939394	24,1236363636364
eps	0,07636363636362	-0,173939393939394	0,07575757575761	0,125454545454545	0,275151515151515	-0,575151515151514	0,374545454545455	-0,275757575757577	-0,226060606060607	0,323636363636364
f2=a2*x*x+a1*x+a0	7,40818181818184	9,30333333333337	11,1871212121213	13,0595454545455	14,9206060606061	16,7703030303031	18,6086363636365	20,4356060606062	22,2512121212123	24,0554545454547
eps2	0,00818181818183561	-0,196666666666633	0,0871212121212643	0,159545454545523	0,320606060606146	-0,529696969696868	0,408636363636482	-0,264393939393802	-0,248787878787724	0,255454545454715

Линейная

Квадратичная

E= 0,836242424242

E= 0,819196969696

etak= 0,575151515151

etak= 0,529696969696

a0= 5,62666666666667

a0= 5,50166666666667

a1= 1,84969696969697

a1= 1,91219696969699

a2= -0,0056818181818

Рис. 3. Результат работы программы квадратичной аппроксимации

Как видно из рис. 3, полученные результаты несколько лучше результатов, полученных при реализации линейной аппроксимации. Следовательно, программа работает верно.

Библиографический список

1. Челебаев С.В. Аппроксимация функциональных зависимостей двух переменных: метод. указ. к практ. работе: Методические указания. Рязань: РИЦ РГРТУ, 2020. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2636>.
2. Челебаев С.В. Методы аппроксимации функций: Методические указания. Рязань: РИЦ РГРТУ, 2019. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2312>.

Лабораторная работа № 1

Изучение метода степенной аппроксимации

Цель работы: изучение метода степенной аппроксимации и его реализация на языке программирования C#

Задание

Найти приближенную функциональную зависимость методом кубической аппроксимации и аппроксимации полиномом 4-й степени для функции. Определить значения параметров аппроксимирующей функции.

Решение. Реализация кубической аппроксимации на языке C#

Рассмотрим ключевые изменения и добавления по сравнению с программой для квадратичной аппроксимации.

// определение функции вычисления определителя 4-го порядка

```
private double opred4(double a11, double a12, double a13, double a14,
    double a21, double a22, double a23, double a24,
    double a31, double a32, double a33, double a34,
    double a41, double a42, double a43, double a44)
{
    return a11 * opred3(a22, a23, a24, a32, a33, a34, a42, a43, a44)
        - a21 * opred3(a12, a13, a14, a32, a33, a34, a42, a43, a44)
        + a31 * opred3(a12, a13, a14, a22, a23, a24, a42, a43, a44)
        - a41 * opred3(a12, a13, a14, a22, a23, a24, a32, a33, a34);
}
```

```

}

// вычисление сумм
s1 = 0.0; // сумма xi
s2 = 0.0; // сумма xi*xi
s3 = 0.0; // сумма xi*xi*xi
s4 = 0.0; // сумма xi*xi*xi*xi
s5 = 0.0; // сумма xi*xi*xi*xi*xi
s6 = 0.0; // сумма xi*xi*xi*xi*xi*xi
s7 = 0.0; // сумма yi
s8 = 0.0; // сумма xi*yi
s9 = 0.0; // сумма xi*xi*yi
s10 = 0.0; // сумма xi*xi*xi*yi
for (int i = 0; i < 10; i++)
{
    s1 += x[i];
    s2 += x[i] * x[i];
    s3 += x[i] * x[i] * x[i];
    s4 += x[i] * x[i] * x[i] * x[i];
    s5 += x[i] * x[i] * x[i] * x[i] * x[i];
    s6 += x[i] * x[i] * x[i] * x[i] * x[i] * x[i];
    s7 += y[i];
    s8 += x[i] * y[i];
    s9 += x[i] * x[i] * y[i];
    s10 += x[i] * x[i] * x[i] * y[i];
}

// вычисление определителей
d = opred4(10,s1,s2,s3,
    s1,s2,s3,s4,
    s2,s3,s4,s5,
    s3,s4,s5,s6);
da0 = opred4(s7,s1,s2,s3,
    s8,s2,s3,s4,
    s9,s3,s4,s5,
    s10,s4,s5,s6);
da1 = opred4(10,s7,s2,s3,
    s1,s8,s3,s4,
    s2,s9,s4,s5,
    s3,s10,s5,s6);
da2 = opred4(10,s1,s7,s3,
    s1,s2,s8,s4,
    s2,s3,s9,s5,
    s3,s4,s10,s6);
da3 = opred4(10,s1,s2,s7,
    s1,s2,s3,s8,
    s2,s3,s4,s9,
    s3,s4,s5,s10);

```

Реализация аппроксимации полиномом 4-й степени на языке C#

Рассмотрим ключевые изменения и добавления по сравнению с программой для кубической аппроксимации.

// определение функции вычисления определителя 5-го порядка

```
private double opred5(double a11, double a12, double a13, double a14, double a15,
    double a21, double a22, double a23, double a24, double a25,
    double a31, double a32, double a33, double a34, double a35,
    double a41, double a42, double a43, double a44, double a45,
    double a51, double a52, double a53, double a54, double a55)
{
    return a11 * opred4(a22, a23, a24, a25,
        a32, a33, a34, a35,
        a42, a43, a44, a45,
        a52, a53, a54, a55)
        - a21 * opred4(a12, a13, a14, a15,
            a32, a33, a34, a35,
            a42, a43, a44, a45,
            a52, a53, a54, a55)
        + a31 * opred4(a12, a13, a14, a15,
            a22, a23, a24, a25,
            a42, a43, a44, a45,
            a52, a53, a54, a55)
        - a41 * opred4(a12, a13, a14, a15,
            a22, a23, a24, a25,
            a32, a33, a34, a35,
            a52, a53, a54, a55)
        + a51 * opred4(a12, a13, a14, a15,
            a22, a23, a24, a25,
            a32, a33, a34, a35,
            a42, a43, a44, a45);
}
```

// вычисление сумм

```
for (int i = 0; i < 10; i++)
{
    s1 += x[i];
    s2 += x[i] * x[i];
    s3 += x[i] * x[i] * x[i];
    s4 += x[i] * x[i] * x[i] * x[i];
    s5 += x[i] * x[i] * x[i] * x[i] * x[i];
    s6 += x[i] * x[i] * x[i] * x[i] * x[i] * x[i];
    s7 += x[i] * x[i] * x[i] * x[i] * x[i] * x[i] * x[i];
    s8 += x[i] * x[i] * x[i] * x[i] * x[i] * x[i] * x[i] * x[i];
    s9 += y[i];
    s10 += x[i] * y[i];
    s11 += x[i] * x[i] * y[i];
    s12 += x[i] * x[i] * x[i] * y[i];
    s13 += x[i] * x[i] * x[i] * x[i] * y[i];
}
```

// вычисление определителей

```

d = opred5(10.0, s1, s2, s3, s4,
s1, s2, s3, s4, s5,
s2, s3, s4, s5, s6,
s3, s4, s5, s6, s7,
s4, s5, s6, s7, s8);
da0 = opred5(s9, s1, s2, s3, s4,
s10, s2, s3, s4, s5,
s11, s3, s4, s5, s6,
s12, s4, s5, s6, s7,
s13, s5, s6, s7, s8);
da1 = opred5(10.0, s9, s2, s3, s4,
s1, s10, s3, s4, s5,
s2, s11, s4, s5, s6,
s3, s12, s5, s6, s7,
s4, s13, s6, s7, s8);
da2 = opred5(10.0, s1, s9, s3, s4,
s1, s2, s10, s4, s5,
s2, s3, s11, s5, s6,
s3, s4, s12, s6, s7,
s4, s5, s13, s7, s8);
da3 = opred5(10.0, s1, s2, s9, s4,
s1, s2, s3, s10, s5,
s2, s3, s4, s11, s6,
s3, s4, s5, s12, s7,
s4, s5, s6, s13, s8);
da4 = opred5(10.0, s1, s2, s3, s9,
s1, s2, s3, s4, s10,
s2, s3, s4, s5, s11,
s3, s4, s5, s6, s12,
s4, s5, s6, s7, s13);

```

Результат работы программы показан на рис. 4.

Form1

	1	2	3	4	5	6	7	8	9	10
x	1	2	3	4	5	6	7	8	9	10
y	7.4	9.5	11.1	12.9	14.6	17.3	18.2	20.7	22.5	23.8
f=a1*x+a0	7.47636363636364	9.32606060606061	11.1757575757576	13.0254545454545	14.8751515151515	16.7248484848485	18.5745454545455	20.4242424242424	22.2739393939394	24.1236363636364
eps	0.07636363636363...	-0.17393939393939...	0.075757575757576	0.125454545454545	0.275151515151515	-0.575151515151514	0.374545454545455	-0.275757575757577	-0.226060606060607	0.323636363636364
f2=a2*x^...	7.408181818181814	9.30333333333337	11.1871212121213	13.0595454545455	14.9206060606061	16.7703030303031	18.6086363636365	20.4356060606062	22.2512121212123	24.0554545454547
eps2	0.00818181818183...	-0.19666666666666...	0.0871212121212643	0.159545454545523	0.320606060606146	-0.529696969696968	0.408636363636482	-0.2643939393939802	-0.248787878787724	0.2554545454545715
f3=a3*x^...	7.53496503496519	9.26107226107293	11.0814685314701	12.9659673659703	14.8843822843871	16.8065268065341	18.7022144522248	20.5412587412729	22.2934731934919	23.9286713286953
eps3	0.134965034965191	-0.2389277389270...	-0.01853146852989...	0.0659673659703142	0.284382284387123	-0.493473193465899	0.502214452224838	-0.158741258727105	-0.206526806508148	0.128671328695287
f4=a4*x^...	7.43741258843003	9.3803030400583	11.1736014336822	12.9497087113088	14.7868300114873	16.7089746693087	18.685956216406	20.633923809542	22.4127050876701	23.8311204578127
eps4	0.03741258843002...	-0.1196969599417...	0.0736014336821782	0.0497087113088188	0.186830011487324	-0.591025330691263	0.485956216406048	-0.0666076190457794	-0.0872949123298667	0.0311204578126585

Линейная

Квадратичная

Кубическая

Полном 4-й степени

E=	0.83624242424242	E=	0.81919696969696	E=	0.74101398601399	E=	0.65701048951522
epsax=	0.57515151515151	epsax=	0.52969696969696	epsax=	0.50221445222483	epsax=	0.59102533069126
a0=	5.62666666666667	a0=	5.50166666666667	a0=	5.93333333333333	a0=	5.15833333333218
a1=	1.84969696969697	a1=	1.91219696969699	a1=	1.52933177933178	a1=	2.52292152285222
		a2=	-0.00568181818181	a2=	0.07733100233114	a2=	-0.28623251771895
				a3=	-0.00503108003107	a3=	0.04464840844204
						a4=	-0.00225815847741

Рис. 4. Результат работы программы степенной аппроксимации

Как видно из рис. 4, полученные результаты несколько лучше результатов, полученных при реализации линейной и квадратичной аппроксимации. Следовательно, программа работает верно.

Контрольные вопросы

1. Расчет параметров кубической аппроксимации.
2. Расчет параметров аппроксимации полиномом 4-й степени.
3. Блок-схема алгоритма расчета кубической аппроксимации.
4. Блок-схема алгоритма расчета аппроксимации полиномом 4-й степени.
5. Вычисление кубической аппроксимации на языке C#.
6. Вычисление аппроксимации полиномом 4-й степени на языке C#.

Содержание отчета

1. Название лабораторной работы.
2. Цель лабораторной работы.
3. Теоретические сведения.
4. Блок-схема алгоритма вычисления кубической аппроксимации.
5. Текст подпрограммы кубической аппроксимации.
6. Текст подпрограммы аппроксимации полиномом 4-й степени.
7. Результаты работы подпрограмм кубической аппроксимации и аппроксимации полиномом 4-й степени.
8. Анализ полученных результатов.

Библиографический список

1. Челебаев С.В. Аппроксимация функциональных зависимостей двух переменных: метод. указ. к практ. работе: Методические указания. Рязань: РИЦ РГРТУ, 2020. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2636>.
2. Челебаев С.В. Методы аппроксимации функций: Методические указания. Рязань: РИЦ РГРТУ, 2019. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2312>.

Практическая работа № 3

Экспоненциальная аппроксимация

Цель работы: изучение метода экспоненциальной аппроксимации и его реализация на языке программирования C#.

Задание. Пусть при изучении неизвестной функциональной зависимости между x и y , полученной в результате серии экспериментов, была сформирована таблица значений (табл. 1). Необходимо найти приближенную функциональную зависимость методом экспоненциальной аппроксимации и определить значения параметров аппроксимирующей функции. Сделать выводы об эффективности экспоненциальной аппроксимации по сравнению с полиномиальной аппроксимацией.

Таблица 1 – Экспериментальная зависимость

x_i	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0	10,0
y_i	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8

Решение

Пример результата работы программы по расчету экспоненциальной аппроксимации показан на рисунке 1.

	1	2	3	4	5	6	7	8	9	10
x	1	2	3	4	5	6	7	8	9	10
y	8	9,7	12,1	14,2	16,3	18,4	20,1	22,2	23,3	24,7
f=a1*x+a0	8,2872727272727	10,2012121212121	12,1151515151515	14,0290909090909	15,9430303030303	17,8569696969696	19,7709090909090	21,6848484848484	23,5987878787878	25,5127272727272
eps	0,2872727272727	0,5012121212121	0,0151515151515	-0,1709090909090	-0,3569696969696	-0,5430303030303	-0,3290909090909	-0,5151515151515	0,2987878787878	0,8127272727272
f2=a2*x*x+a1*x+a0	7,6599999999999	9,9921212121212	12,2196969696969	14,3427272727272	16,3612121212121	18,2751515151515	20,0845454545454	21,7893939393939	23,3896969696969	24,8854545454545
eps2	-0,3400000000000	0,2921212121212	0,1196969696969	0,1427272727272	0,0612121212121	-0,1248484848484	-0,0154545454545	-0,4106060606060	0,0896969696969	0,1854545454545
f3=a3*x*x*x+a2*x	7,907692307770	9,909557109693	12,01328671348	14,15990676017	16,29044289078	18,34592074633	20,267365967853	21,99580419636	23,47226107290	24,63776223847
eps3	-0,09230769222	0,209557109693	-0,08671328651	-0,04009323982	-0,00955710921	-0,05407925366	0,167365967853	-0,20419580363	0,172261072900	-0,06223776152
f4=a4*x*x*x*x+a3*x	7,951748230291	9,855710975353	11,97167840199	14,16724957947	16,33449910226	18,38997709004	20,27470918769	21,954196565323	23,41841591822	24,68181946692
eps4	-0,04825176970	0,155710975353	-0,12832159800	-0,03275042052	0,034499102264	-0,01002290995	0,174709187696	-0,24580343467	0,118415918227	-0,01818053307
f5=a5*x^5+(k*x)+b	8,410112879645	10,24665803030	12,10166076658	13,975306590312	15,86778286762	17,77927884771	19,70989568179	21,66009644214	23,62980614147	25,61931175238
eps5	0,410112879645	0,546658030306	0,001660766588	-0,22469340968	-0,43221713237	-0,62072115228	-0,39001431820	-0,53990355785	0,329806141475	0,919311752387

E = 1.9089696969697
emax = 0.812727272727269
a0 = 6.37333333333331
a1 = 1.9139393939394

Линейная аппроксимация

E = 0.4662424242425
emax = 0.41060606060513
a0 = 5.2233333333332
a1 = 2.4889393939395
a2 = -0.05227272727294

Квадратичная аппроксимация

E = 0.167832167832166
emax = 0.209557109693028
a0 = 6.066666666669307
a1 = 1.74094794099621
a2 = 0.109906759910302
a3 = -0.00982905982920797

Кубическая аппроксимация

E = 0.150699300703113
emax = 0.245803434676951
a0 = 6.41666662275584
a1 = 1.29223000548777
a2 = 0.274096745791094
a3 = -0.0322649572935051
a4 = 0.00101981355036856

Аппроксимация полинома 4-ой степени

E = 2.48713707027951
emax = 0.919311752387546
a = 180.919501721379
b = -174.327660062818

Экспоненциальная аппроксимация

Рисунок 1 – Результат работы программы аппроксимации

Текст программы.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static double opred2(double a11, double a12, double a21, double a22) //Вычисление определителя 2*2
        {
            return a11 * a22 - a12 * a21;
        }

        static void Main(string[] args)
        {
            // объявление переменных
            double[] x = new double[10]; // массив значений xi
            double[] y = new double[10]; // массив значений yi
            double s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s12, s13; // суммы
            double[] f = new double[10]; // массив аппроксимированных значений fi
            double[] eps = new double[10]; // массив погрешностей ei
            double E; // сумма квадратов погрешностей
            double emax; // максимальная абсолютная погрешность
            double d; // главный определитель
            double da0, da1, da2, da3, da4, da, db; //определители
            double a0, a1, a2, a3, a4, a, b; //параметры аппроксимации
            double k;

            x[0] = 1.0; y[0] = 7.4;
            x[1] = 2.0; y[1] = 9.5;

```

```

x[2] = 3.0; y[2] = 11.1;
x[3] = 4.0; y[3] = 12.9;
x[4] = 5.0; y[4] = 14.6;
x[5] = 6.0; y[5] = 17.3;
x[6] = 7.0; y[6] = 18.2;
x[7] = 8.0; y[7] = 20.7;
x[8] = 9.0; y[8] = 22.5;
x[9] = 10.0; y[9] = 24.8;

for (k = 0.2; k < 2.0; k += 0.01)
{ // искать также в отрицательной области
    //Вычисление сумм
    s1 = 0.0; // сумма  $e^{2 \cdot x_i}$ 
    s2 = 0.0; // сумма  $e^{x_i}$ 
    s3 = 0.0; // сумма  $e^{x_i} \cdot y_i$ 
    s4 = 0.0; // сумма  $y_i$ 
    for (int i = 0; i < 10; i++)
    {
        s1 += Math.Exp(2 * k * x[i]);
        s2 += Math.Exp(k * x[i]);
        s3 += Math.Exp(k * x[i]) * y[i];
        s4 += y[i];
    }
    // вычисление определителей
    d = opred2(s1, s2, s2, 10);
    da = opred2(s3, s2, s4, 10);
    db = opred2(s1, s3, s2, s4);
    a = da / d;
    b = db / d;
    E = 0.0;
    emax = 0.0;
    for (int i = 0; i < 10; i++)
    {
        f[i] = a * Math.Exp(x[i]) + b;
        eps[i] = f[i] - y[i];
        E += eps[i] * eps[i];
        if (Math.Abs(eps[i]) > emax)
            emax = Math.Abs(eps[i]);
    }
    /*
        for (int i = 0; i < 10; i++)
        {
            Console.WriteLine("x=" + x[i] + " y=" + y[i] + " f=" + f[i] + "
eps=" + eps[i]);
        } */
    Console.WriteLine("k=" + k + " E=" + E + " emax=" + emax + " a=" + a + " b=" +
b);
}
Console.ReadKey();
} // end main
}

```

Контрольные вопросы

1. Экспоненциальная аппроксимация с использованием трех параметров.
2. Экспоненциальная аппроксимация с использованием двух параметров.
3. Расчет параметров экспоненциальной аппроксимации.
4. Блок-схема алгоритма расчета экспоненциальной аппроксимации.
5. Вычисление экспоненциальной аппроксимации на языке C#.

Варианты заданий

Варианты заданий для выполнения лабораторной работы совпадают с практической работой № 1.

Библиографический список

1. Челебаев С.В. Аппроксимация функциональных зависимостей двух переменных: метод. указ. к практ. работе: Методические указания. Рязань: РИЦ РГРТУ, 2020. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2636>.

2. Челебаев С.В. Методы аппроксимации функций: Методические указания. Рязань: РИЦ РГРТУ, 2019. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2312>.

Практическая работа № 4

Логарифмическая аппроксимация

Цель работы: изучение метода логарифмической аппроксимации и его реализация на языке программирования C#.

Задание. Пусть при изучении неизвестной функциональной зависимости между x и y , полученной в результате серии экспериментов, была сформирована таблица значений (табл. 1). Необходимо найти приближенную функциональную зависимость методом логарифмической аппроксимации и определить значения параметров аппроксимирующей функции. Сделать выводы об эффективности логарифмической аппроксимации по сравнению с полиномиальной и экспоненциальной аппроксимацией.

Таблица 1 – Экспериментальная зависимость

x_i	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0	10,0
y_i	7,4	9,5	11,1	12,9	14,6	17,3	18,2	20,7	22,5	23,8

Решение

Пример результата работы программы по расчету логарифмической аппроксимации показан на рисунке 1.

	1	2	3	4	5	6	7	8	9	10
x	1	2	3	4	5	6	7	8	9	10
y	8	9,7	12,1	14,2	16,3	18,4	20,1	22,2	23,3	24,7
f=a1*x+a0	8,2872727272727	10,20121212121...	12,11515151515...	14,02909090909...	15,94303030303...	17,85696969696...	19,77090909090...	21,68484848484...	23,59878787878...	25,51272727272...
eps	0,2872727272727	0,501212121212...	0,015151515151...	-0,170909090909...	-0,35696969696...	-0,54303030303...	-0,32909090909...	-0,51515151515...	0,29878787878...	0,81272727272...
f2=a2*x*x+a1*x+a0	7,6599999999999	9,992121212121...	12,21969696969...	14,34272727272...	16,36121212121...	18,27515151515...	20,08454545454...	21,78939393939...	23,38969696969...	24,88545454545...
eps2	-0,34000000000...	0,292121212121...	0,11969696969...	0,14272727272...	0,06121212121...	-0,12484848484...	-0,01545454545...	-0,41060606060...	0,08969696969...	0,18545454545...
f3=a3*x*x*x+a2*x...	7,907692307770...	9,90957109693...	12,01328671348...	14,15990676017...	16,29044289078...	18,34592074633...	20,267365967853...	21,99580419636...	23,47226107290...	24,63776223847...
eps3	-0,09230769222...	0,209557109693...	-0,08671328651...	-0,04009323982...	-0,00955710921...	-0,05407925366...	0,167365967853...	-0,20419580363...	0,17226107290...	-0,06223776152...
f4=a4*x*x*x*x+a3...	7,951748230291...	9,855710975353...	11,97167840199...	14,16274957947...	16,33449910226...	18,38997709004...	20,27470918769...	21,954196565323...	23,41841591822...	24,68181946692...
eps4	-0,04825176970...	0,155710975353...	-0,12832159800...	-0,03275042052...	0,034499102264...	-0,01002290995...	0,174709187696...	-0,24580343467...	0,11841591822...	-0,01818053307...
f5=a5*x^(k*x)+b	8,410112879645...	10,2465803030...	12,10166076658...	13,975306590312...	15,86778286762...	17,77927884771...	19,70998568179...	21,66009644214...	23,62980614147...	25,61931175238...
eps5	0,410112879645...	0,546658030306...	0,001660766588...	-0,22469340968...	-0,43221713237...	-0,62072115228...	-0,39001431820...	-0,53990355785...	0,329806141475...	0,919311752387...
f6=a7*n(x+1)+b	6,077003793629...	11,02244203901...	14,00108886558...	16,13936845766...	17,80858310118...	19,17795330337...	20,33897359859...	21,34675189884...	22,23706433450...	23,03445797326...
* eps6	-1,92299620637...	1,322442039016...	1,901088865580...	1,939368457669...	1,508583101184...	0,777953303377...	0,238973598595...	-0,85324810115...	-1,06293566549...	-1,66554202673...

E = 1.9089696969697

eps = 0.812727272727269

a0 = 6.37333333333331

a1 = 1.9139393939394

Линейная аппроксимация

E = 0.4662424242425

eps = 0.4106060606060513

a0 = 5.2233333333332

a1 = 2.4889393939395

a2 = -0.05227272727294

Квадратичная аппроксимация

E = 0.167832167832166

eps = 0.209557109693028

a0 = 6.0666666666669307

a1 = 1.74094794099621

a2 = 0.109906759910302

a3 = -0.00982905982920797

Кубическая аппроксимация

E = 0.150699300703113

eps = 0.245803434676951

a0 = 6.41666662275584

a1 = 1.29223000548777

a2 = 0.274096745791094

a3 = -0.0322649572935051

a4 = 0.00101981355036856

Аппроксимация полинома 4 - ой степени

E = 2.48713707027951

eps = 0.919311752387546

a = 180.919501721379

b = -174.327660062818

Экспоненциальная аппроксимация

E = 20.392093735856

eps = 1.66554202673082

a = 7.64805209808973

c = -12.2622241784956

Логарифмическая аппроксимация

Рисунок 1 – Результат работы программы аппроксимации

Контрольные вопросы

1. Логарифмическая аппроксимация с использованием трех параметров.
2. Логарифмическая аппроксимация с использованием двух параметров.
3. Расчет параметров логарифмической аппроксимации.
4. Блок-схема алгоритма расчета логарифмической аппроксимации.
5. Вычисление логарифмической аппроксимации на языке C#.

Варианты заданий

Варианты заданий для выполнения работы № 2 совпадают с практической работой № 1.

Содержание отчета

1. Название лабораторной работы.
2. Цель лабораторной работы.
3. Теоретические сведения.
4. Блок-схема алгоритма вычисления логарифмической аппроксимации.
5. Текст подпрограммы логарифмической аппроксимации.
6. Результаты работы подпрограмм полиномиальной, экспоненциальной и логарифмической аппроксимации.
7. Анализ полученных результатов.

Библиографический список

1. Челебаев С.В. Аппроксимация функциональных зависимостей двух переменных: метод. указ. к практ. работе: Методические указания. Рязань: РИЦ РГРТУ, 2020. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2636>.
2. Челебаев С.В. Методы аппроксимации функций: Методические указания. Рязань: РИЦ РГРТУ, 2019. — Режим доступа: <https://elib.rsreu.ru/ebs/download/2312>.

Лабораторная работа № 2

Быстрое преобразование Фурье

Цель работы: реализация быстрого преобразования Фурье на языке программирования высокого уровня.

Задание. Реализовать 32-точечное БПФ.

БПФ рассчитать для следующих функций (использовать 32 отсчета):

- 1) синус (в отсчеты должно уместиться 2 периода);
- 2) косинус (в отсчеты должно уместиться 2 периода);
- 3) синус (2 составляющие);
- 4) косинус (2 составляющие);
- 5) синус (3 составляющие);
- 6) косинус (3 составляющие);
- 7) синус + косинус.

Контрольные вопросы

1. Понятие быстрого преобразования Фурье.
2. Разделение исходной последовательности прореживанием по времени.
3. Процедура объединения.
4. Граф «бабочка».
5. Полный граф алгоритма БПФ с прореживанием по времени. Двоично-инверсная перестановка.

2. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ СТУДЕНТАМ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Перед началом изучения дисциплины студенту необходимо ознакомиться с содержанием рабочей программы дисциплины, с целями и задачами дисциплины, ее связями с другими дисциплинами образовательной программы, методическими разработками по данной дисциплине, имеющимися на образовательном портале РГРТУ и сайте кафедры.

Методические рекомендации студентам по работе над конспектом лекции

Основу теоретического обучения студентов составляют лекции. Они дают систематизированные знания студентам о наиболее сложных и актуальных проблемах изучаемой дисциплины. На лекциях особое внимание уделяется не только усвоению студентами изучаемых проблем, но и стимулированию их активной познавательной деятельности, творческого мышления, развитию научного мировоззрения, профессионально-значимых свойств и качеств.

Перед каждой лекцией студенту необходимо просматривать рабочую программу дисциплины, что позволит сэкономить время на записывание темы лекции, ее основных вопросов, рекомендуемой литературы.

Перед очередной лекцией необходимо просмотреть по конспекту материал предыдущей лекции. При затруднениях в восприятии материала следует обратиться к основным литературным источникам. Если разобраться в материале опять не удалось, то обратитесь к лектору (по графику его консультаций) или к преподавателю на практических занятиях. Не оставляйте «белых пятен» в освоении материала.

Во время лекции студенты должны не только внимательно воспринимать действия преподавателя, но и самостоятельно мыслить, добиваться понимания изучаемого предмета. Студенты должны аккуратно вести конспект. В случае недопонимания какой-либо части предмета следует задать вопрос в установленном порядке преподавателю. В процессе работы на лекции необходимо так же выполнять в конспектах модели изучаемого предмета (рисунки, схемы, чертежи и т.д.), которые использует преподаватель.

Слушая лекцию, нужно из всего получаемого материала выбирать и записывать самое главное. Следует знать, что главные положения лекции преподаватель обычно выделяет интонацией или повторяет несколько раз. Именно поэтому предварительная подготовка к лекции позволит студенту уловить тот момент, когда следует перейти к конспектированию, а когда можно просто внимательно слушать лекцию. В связи с этим нелишне перед началом сессии еще раз бегло просмотреть учебники или прежние конспекты по изучаемым предметам. Это станет первичным знакомством с тем материалом, который прозвучит на лекции, а также создаст необходимый психологический настрой.

Чтобы правильно и быстро конспектировать лекцию важно учитывать, что способы подачи лекционного материала могут быть разными. Преподаватель может диктовать материал, или рассказывать его, не давая ничего под запись, или проводить занятие в форме диалога со студентами. Чаще всего можно наблюдать соединение двух или трех вышеперечисленных способов.

Эффективность конспектирования зависит от умения владеть правильной методикой записи лекции. Конечно, способы конспектирования у каждого человека индивидуальны. Однако существуют некоторые наиболее употребляемые и целесообразные приемы записи лекционного материала.

Запись лекции можно вести в виде тезисов – коротких, простых предложений, фиксирующих только основное содержание материала. Количество и краткость тезисов может определяться как преподавателем, так и студентом. Естественно, что такая запись лекции требует впоследствии обращения к дополнительной литературе. На отдельные лекции можно приносить соответствующий иллюстративный материал на бумажных или электронных носителях, представленный лектором на портале или присланный на «электронный почтовый ящик группы» (таблицы, графики, схемы). Данный материал будет охарактеризован, прокомментирован, дополнен непосредственно на лекции.

Кроме тезисов важно записывать примеры, доказательства, даты и цифры. Значительно облегчают понимание лекции те схемы и графики, которыми преподаватель иллюстрирует теоретический материал. По мере возможности студенты должны переносить их в тетрадь рядом с тем текстом, к которому эти схемы и графики относятся.

Хорошо если конспект лекции дополняется собственными мыслями, суждениями, вопросами, возникающими в ходе прослушивания содержания лекции. Те вопросы, которые возникают у студента при конспектировании лекции, не всегда целесообразно задавать сразу при их возникновении, чтобы не нарушить ход рассуждений преподавателя. Студент может попытаться ответить на них сам в процессе подготовки к практическим занятиям либо обсудить их с преподавателем на консультации.

Важно и то, как будет расположен материал в лекции. Если запись тезисов ведется по всей строке, то целесообразно отделять их время от времени красной строкой или пропуском строки. Примеры же и дополнительные сведения можно смещать вправо или влево под тезисом, а также на поля. В тетради нужно выделять темы лекций, записывать рекомендуемую для самостоятельной подготовки литературу, внести фамилию, имя и отчество преподавателя. Наличие полей в тетради позволяет не только получить «ровный» текст, но и дает возможность при необходимости вставить важные дополнения и изменения в конспект лекции.

При составлении конспектов необходимо использовать избыточность русского языка, сокращая слова. Так в процессе совершенствования навыков конспектирования лекций важно выработать индивидуальную систему записи материала, научиться рационально сокращать слова и отдельные словосочетания.

Практика показывает, что не всегда студенту удастся успевать записывать слова лектора даже при использовании приемов сокращения слов. В этом случае допустимо обратиться к лектору с просьбой повторить сказанное. При обращении важно четко сформулировать просьбу, указать какой отрывок необходимо воспроизвести еще раз. Однако не всегда удобно прерывать ход лекции. В этом случае можно оставить пропуск, и после лекции устранить его при помощи конспекта соседа. Важно сделать это в короткий срок, пока свежа память о воспринятой на лекции информации.

Работу над конспектом следует начинать с его доработки, желательно в тот же день, пока материал еще легко воспроизводим в памяти (через 10 часов после лекции в памяти остается не более 30-40 % материала). С целью доработки необходимо прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Далее следует прочитать материал по рекомендуемой литературе, разрешая в ходе чтения возникшие ранее затруднения, вопросы, а также дополняя и исправляя свои

записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект.

Подготовленный конспект и рекомендуемая литература используются при подготовке к лабораторным работам и практическим занятиям. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы. Примеры, задачи, вопросы по теме являются средством самоконтроля.

Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой степени требуется подготовить к очередному занятию. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний.

Методические рекомендации студентам по работе с литературой

В рабочей программе дисциплины для каждого раздела и темы дисциплины указывается основная и дополнительная литература, позволяющая более глубоко изучить данный вопрос. Обычно список всей рекомендуемой литературы преподаватель озвучивает на первой лекции или дает ссылки на ее местонахождение (на образовательном портале РГРТУ, на сайте кафедры и т.д.).

При работе с рекомендуемой литературой целесообразно придерживаться такой последовательности. Сначала лучше прочитать заданный текст в быстром темпе. Цель такого чтения заключается в том, чтобы создать общее представление об изучаемом материале, понять общий смысл прочитанного. Затем прочитать вторично, более медленно, чтобы в ходе чтения понять и запомнить смысл каждой фразы, каждого положения и вопроса в целом.

Чтение приносит пользу и становится продуктивным, когда сопровождается записями. Это может быть составление плана прочитанного текста, тезисы или выписки, конспектирование и др. Выбор вида записи зависит от характера изучаемого материала и целей работы с ним. Если содержание материала несложное, легко усваиваемое, можно ограничиться составлением плана. Если материал содержит новую и трудно усваиваемую информацию, целесообразно его законспектировать.

План – это схема прочитанного материала, перечень вопросов, отражающих структуру и последовательность материала.

Конспект – это систематизированное, логичное изложение материала источника. Различаются четыре типа конспектов:

- план-конспект – это развернутый детализированный план, в котором по наиболее сложным вопросам даются подробные пояснения,
- текстуальный конспект – это воспроизведение наиболее важных положений и фактов источника,
- свободный конспект – это четко и кратко изложенные основные положения в результате глубокого изучения материала, могут присутствовать выписки, цитаты, тезисы; часть материала может быть представлена планом,
- тематический конспект – составляется на основе изучения ряда источников и дает ответ по изучаемому вопросу.

В процессе изучения материала источника и составления конспекта нужно обязательно применять различные выделения, подзаголовки, создавая блочную структуру конспекта. Это делает конспект легко воспринимаемым и удобным для работы.

Методические рекомендации студентам по подготовке к лабораторным работам

Лабораторная работа — это форма организации учебного процесса, когда обучающиеся по заданию и под руководством преподавателя самостоятельно проводят вычислительные расчеты и экспериментальные исследования на основе специально разработанных заданий.

Для проведения лабораторных работ используется вычислительная техника, которая размещается в специально оборудованных учебных лабораториях. Перед началом цикла лабораторных работ преподаватель или другое ответственное лицо проводит с обучающимися инструктаж о правилах техники безопасности в данной лаборатории, после чего студенты расписываются в специальном журнале техники безопасности.

По каждой лабораторной работе разрабатываются методические указания по их проведению. Они используются обучающимися при выполнении лабораторной работы.

Применяются разные формы организации обучающихся на лабораторных работах: фронтальная, групповая и индивидуальная. При фронтальной форме организации занятий все обучающиеся выполняют одновременно одну и ту же работу. При групповой форме организации занятий одна и та же работа выполняется группами по 2-5 человек. При индивидуальной форме организации занятий каждый обучающийся выполняет индивидуальное задание. Выбор метода зависит от учебно-методической базы и задач дисциплины.

До начала лабораторной работы студент должен ознакомиться с теоретическими вопросами, которые будут изучаться или исследоваться в этой работе. Также необходимо познакомиться с принципами работы лабораторного оборудования, используемого в лабораторной работе. Перед началом лабораторной работы преподаватель может провести проверку знаний обучающихся - их теоретической готовности к выполнению задания. По итогам этой проверки студент допускается или не допускается к данной работе. О такой исходной проверке преподаватель информирует студентов заранее. Также возможна ситуация, когда допуском к очередной лабораторной работе является своевременная сдача предыдущей лабораторной работы (или подготовка отчета по ней).

Во время лабораторной работы обучающиеся выполняют запланированное лабораторное задание. Все полученные результаты необходимо зафиксировать в черновике отчета или сохранить в электронном виде на сменном носителе.

Завершается лабораторная работа оформлением индивидуального отчета и его защитой перед преподавателем.

Методические рекомендации студентам по подготовке к экзамену

При подготовке к экзамену студент должен повторно изучить конспекты лекций и рекомендованную литературу, просмотреть решения основных задач, решенных самостоятельно и на практических занятиях.

Необходимо помнить, что промежутки между очередными экзаменами обычно составляют всего несколько дней. Поэтому подготовку к ним нужно начинать заблаговременно в течение семестра. До наступления сессии уточните у преподавателя порядок проведения промежуточной аттестации по его предмету и формулировки критериев для количественной оценивания уровня подготовки студентов. Для итоговой положительной оценки по предмету необходимо вовремя и с нужным качеством выполнить или защитить контрольные работы, лабораторные работы, так как всё это может являться обязательной частью учебного процесса по данной дисциплине.

Рекомендуется разработать план подготовки к каждому экзамену, в котором указать, какие вопросы или билеты нужно выучить, какие задачи решить за указанный в плане временной отрезок.

Также бывает полезно вначале изучить более сложные вопросы, а затем переходить к изучению более простых вопросов. При этом желательно в начале каждого следующего дня подготовки бегло освежить в памяти выученный ранее материал.

В период экзаменационной сессии организм студента работает в крайне напряженном режиме и для успешной сдачи сессии нужно не забывать о простых, но обязательных правилах:

- по возможности обеспечить достаточную изоляцию: не отвлекаться на разговоры с друзьями, просмотры телепередач, общение в социальных сетях;
- уделять достаточное время сну;
- отказываться от успокоительных. Здоровое волнение – это нормально. Лучше снимать волнение небольшими прогулками, самовнушением;
- внушать себе, что сессия – это не проблема. Это нормальный рабочий процесс. Не накручивайте себя, не создавайте трагедий в своей голове;

- помогите своему организму – обеспечьте ему полноценное питание, давайте ему периоды отдыха с переменной вида деятельности;
- следуйте плану подготовки.

Методические рекомендации студентам по проведению самостоятельной работы

Самостоятельная работа студента над учебным материалом является неотъемлемой частью учебного процесса в вузе.

В учебном процессе образовательного учреждения выделяются два вида самостоятельной работы:

1) аудиторная – выполняется на учебных занятиях, под непосредственным руководством преподавателя и по его заданию), студентам могут быть предложены следующие виды заданий:

- выполнение самостоятельных работ;
- выполнение лабораторных работ;
- составление схем, диаграмм, заполнение таблиц;
- решение задач;
- работу со справочной, нормативной документацией и научной литературой;
- защиту выполненных работ;
- тестирование и т.д.

2) внеаудиторная – выполняется по заданию преподавателя, но без его непосредственного участия, включает следующие виды деятельности.

- подготовку к аудиторным занятиям (теоретическим и лабораторным работам);
- изучение учебного материала, вынесенного на самостоятельную проработку: работа над определенными темами, разделами, вынесенными на самостоятельное изучение в соответствии с рабочими программами учебной дисциплины или профессионального модуля;
- выполнение домашних заданий разнообразного характера;
- выполнение индивидуальных заданий, направленных на развитие у студентов самостоятельности и инициативы;
- подготовку к практической работе, экзамену;
- другие виды внеаудиторной самостоятельной работы.

Внеаудиторные самостоятельные работы представляют собой логическое продолжение аудиторных занятий, проводятся по заданию преподавателя, который инструктирует студентов и устанавливает сроки выполнения задания.

При планировании заданий для внеаудиторной самостоятельной работы используются следующие типы самостоятельной работы:

- воспроизводящая (репродуктивная), предполагающая алгоритмическую деятельность по образцу в аналогичной ситуации. Включает следующую основную деятельность: самостоятельное прочтение, просмотр, конспектирование учебной литературы, прослушивание записанных лекций, заучивание, пересказ, запоминание, Internet–ресурсы, повторение учебного материала и др.
- реконструктивная, связанная с использованием накопленных знаний и известного способа действия в частично измененной ситуации, предполагает подготовку отчетов по лабораторным работам, подбор литературы по дисциплинарным проблемам, подготовка к защите лабораторных работ и др.
- эвристическая (частично-поисковая) и творческая, направленная на развитие способностей студентов к исследовательской деятельности.

Одной из важных форм самостоятельной работы студента является работа с литературой ко всем видам занятий. Самостоятельная работа студента с литературой позволяет ему более углубленно вникнуть в изучаемую тему.

Один из методов работы с литературой – повторение: прочитанный текст можно заучить наизусть. Простое повторение воздействует на память механически и поверхностно. Полученные таким путем сведения легко забываются.

Более эффективный метод – метод кодирования: прочитанный текст нужно подвергнуть большей, чем простое заучивание, обработке. Чтобы основательно обработать информацию и закодировать ее для хранения, важно провести целый ряд мыслительных операций: прокомментировать новые данные; оценить их значение; поставить вопросы; сопоставить полученные сведения с ранее известными. Для

улучшения обработки информации очень важно устанавливать осмысленные связи, структурировать новые сведения.

Изучение научной, учебной и иной литературы требует ведения рабочих записей. Форма записей может быть весьма разнообразной: простой или развернутый план, тезисы, цитаты, конспект.

План – структура письменной работы, определяющая последовательность изложения материала. Он является наиболее краткой и потому самой доступной и распространенной формой записей содержания исходного источника информации. По существу, это перечень основных вопросов, рассматриваемых в источнике. План может быть простым и развернутым. Их отличие состоит в степени детализации содержания и, соответственно, в объеме.

Преимущество плана состоит в том, что план позволяет наилучшим образом уяснить логику мысли автора, упрощает понимание главных моментов произведения. Кроме того, он позволяет быстро и глубоко проникнуть в сущность построения произведения и, следовательно, гораздо легче ориентироваться в его содержании и быстрее обычного вспомнить прочитанное. С помощью плана гораздо удобнее отыскивать в источнике нужные места, факты, цитаты и т. д.

Выписки представляют собой небольшие фрагменты текста (неполные и полные предложения, отдельные абзацы, а также дословные и близкие к дословным записи об излагаемых в нем фактах), содержащие в себе основной смысл содержания прочитанного. Выписки представляют собой более сложную форму записи содержания исходного источника информации. По сути, выписки – не что иное, как цитаты, заимствованные из текста. Выписки позволяют в концентрированной форме и с максимальной точностью воспроизвести наиболее важные мысли автора. В отдельных случаях – когда это оправдано с точки зрения продолжения работы над текстом – вполне допустимо заменять цитирование изложением, близким дословному.

Тезисы – сжатое изложение содержания изученного материала в утвердительной (реже опровергающей) форме. Отличие тезисов от обычных выписок состоит в том, что тезисам присуща значительно более высокая степень концентрации материала. В тезисах отмечается преобладание выводов над общими рассуждениями. Записываются они близко к оригинальному тексту, т. е. без использования прямого цитирования.

Аннотация – краткое изложение основного содержания исходного источника информации, дающее о нем обобщенное представление. К написанию аннотаций прибегают в тех случаях, когда подлинная ценность и пригодность исходного источника информации исполнителю письменной работы окончательно неясна, но в то же время о нем необходимо оставить краткую запись с обобщающей характеристикой.

Резюме – краткая оценка изученного содержания исходного источника информации, полученная, прежде всего, на основе содержащихся в нем выводов. Резюме весьма сходно по своей сути с аннотацией. Однако, в отличие от последней, текст резюме концентрирует в себе данные не из основного содержания исходного источника информации, а из его заключительной части, прежде всего, выводов. Но, как и в случае с аннотацией, резюме излагается своими словами – выдержки из оригинального текста в нем практически не встречаются.

Конспект представляет собой сложную запись содержания исходного текста, включающая в себя заимствования (цитаты) наиболее примечательных мест в сочетании с планом источника, а также сжатый анализ записанного материала и выводы по нему.

При выполнении конспекта требуется внимательно прочитать текст, уточнить в справочной литературе непонятные слова и вынести справочные данные на поля конспекта. Нужно выделить главное, составить план. Затем следует кратко сформулировать основные положения текста, отметить аргументацию автора. Записи материала следует проводить, четко следуя пунктам плана и выражая мысль своими словами. Цитаты должны быть записаны грамотно, учитывать лаконичность, значимость мысли.

В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и

дополнения необходимо оставлять поля. Необходимо указывать библиографическое описание конспектируемого источника.

3. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

а) основная литература

1. Реннер А.Г. Аппроксимация функций обобщенным рядом Фурье [Электронный ресурс]: методические указания к лабораторному практикуму и самостоятельной работе студентов/ Реннер А.Г., Корнейченко Д.В.— Электрон. текстовые данные.— Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2014.— 27 с.— Режим доступа: <http://www.iprbookshop.ru/50033.html>.— ЭБС «IPRbooks»

2. Малышева Т.А. Численные методы и компьютерное моделирование. Лабораторный практикум по аппроксимации функций [Электронный ресурс]: учебно-методическое пособие/ Малышева Т.А.— Электрон. текстовые данные.— СПб.: Университет ИТМО, 2016.— 33 с.— Режим доступа: <http://www.iprbookshop.ru/67833.html>.— ЭБС «IPRbooks»

3. Езерский В.В. Избранные разделы высшей математики. Выпуск 7. Методы аппроксимации функций [Электронный ресурс]: учебное пособие/ Езерский В.В.— Электрон. текстовые данные.— Омск: Сибирский государственный университет физической культуры и спорта, 2011.— 52 с.— Режим доступа: <http://www.iprbookshop.ru/64938.html>.— ЭБС «IPRbooks»

4. Барский А.Б. Введение в нейронные сети [Электронный ресурс] / Барский А.Б.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 358 с.— Режим доступа: <http://www.iprbookshop.ru/52144.html>.— ЭБС «IPRbooks»

5. Яхьяева Г.Э. Нечеткие множества и нейронные сети [Электронный ресурс]: учебное пособие/ Яхьяева Г.Э.— Электрон. текстовые данные.— Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017.— 320 с.— Режим доступа: <http://www.iprbookshop.ru/67390.html>.— ЭБС «IPRbooks»

6. Горожанина Е.И. Нейронные сети [Электронный ресурс]: учебное пособие/ Горожанина Е.И.— Электрон. текстовые данные.— Самара: Поволжский государственный университет телекоммуникаций и информатики, 2017.— 84 с.— Режим доступа: <http://www.iprbookshop.ru/75391.html>.— ЭБС «IPRbooks»

7. Аппроксимация функций: методические указания к лабораторным работам / Рязан. гос. радиотехн. ун-т; сост.: С.В. Челебаев. Рязань, 2019. 24 с.

8. Локтюхин В.Н., Челебаев С.В., Антоненко А.В. Нейросетевые преобразователи информации: синтез и программирование на ПЛИС / Научное издание, Рязань, Издательство «Сервис», 2011. 128 с.

9. Чигирёва О.Ю. Ряды Фурье. Преобразование Фурье [Электронный ресурс]: методические указания / Чигирёва О.Ю. — Электрон. текстовые данные. — М.: Московский государственный технический университет имени Н.Э. Баумана, 2010.— 52 с.— Режим доступа: <http://www.iprbookshop.ru/31238.html>.— ЭБС «IPRbooks»

10. Волков В.А. Ряды Фурье. Интегральные преобразования Фурье и Радона [Электронный ресурс]: учебно-методическое пособие/ Волков В.А.— Электрон. текстовые данные.— Екатеринбург: Уральский федеральный университет, 2014.— 32 с.— Режим доступа: <http://www.iprbookshop.ru/66202.html>.— ЭБС «IPRbooks»

11. Высшая математика. Том 4. Дифференциальные уравнения. Ряды. Ряды Фурье и преобразование Фурье. Дифференциальное и интегральное исчисление функций нескольких переменных. Теория поля [Электронный ресурс]: учебник/ А.П. Господариков [и др.].— Электрон. текстовые данные.— СПб.: Санкт-Петербургский горный университет, 2015.— 213 с.— Режим доступа: <http://www.iprbookshop.ru/71690.html>. — ЭБС «IPRbooks»

б) дополнительная литература

1. Веричев С.Н. Специальные главы высшей математики. Руководство к решению задач по теории вероятностей [Электронный ресурс]: учебное пособие/ Веричев С.Н., Икрянников В.И., Бутырин В.И.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2009.— 100 с.— Режим доступа: <http://www.iprbookshop.ru/45437.html>.— ЭБС «IPRbooks»

2. Прохоров Ю.В. Лекции по теории вероятностей и математической статистике [Электронный ресурс]: учебник/ Прохоров Ю.В., Пономаренко Л.С.— Электрон. текстовые данные.— М.: Московский государственный университет имени М.В. Ломоносова, 2012.— 254 с.— Режим доступа: <http://www.iprbookshop.ru/13173.html>.— ЭБС «IPRbooks»

3. Зарипов Р.Н. Специальные разделы математики. Теория функций комплексной переменной. Основы операционного исчисления [Электронный ресурс]: учебное пособие/ Зарипов Р.Н., Чугунова Г.П.— Электрон. текстовые данные.— Казань: Казанский национальный исследовательский технологический университет, 2008.— 115 с.— Режим доступа: <http://www.iprbookshop.ru/63467.html>.— ЭБС «IPRbooks»

4. КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ ПОДГОТОВКИ К ЭКЗАМЕНУ

ОПК-1.1.

1. Классы задач аппроксимации.
2. Сжатие информации.
3. Восстановление функциональной зависимости.
4. Сглаживание экспериментальных данных, фильтрация помех.
5. Классы приближающих функций.
6. Приближение многочленами.
7. Класс тригонометрических полиномов.
8. Обобщенный полином.
9. Ортогональные системы.
10. Класс рациональных дробей.
11. Класс экспоненциальной суммы.
12. Аппарат сплайн-функций.
13. Фрактальные методы аппроксимации.
14. Линейная аппроксимация.
15. Метод наименьших квадратов.
16. Метод наименьших модулей.
17. Расчет коэффициентов линейной аппроксимации.
18. Кусочно-линейная аппроксимация.
19. Квадратичная аппроксимация.
20. Расчет коэффициентов квадратичной аппроксимации.
21. Реализация квадратичной аппроксимации на языке программирования высокого уровня.
22. Полиномиальная аппроксимация.
23. Расчет коэффициентов аппроксимации полиномами 3-й и 4-й степени.
24. Реализация аппроксимации полиномами 3-й и 4-й степени на языке программирования высокого уровня.
25. Аппроксимация функциональных зависимостей двух переменных.
26. Экспоненциальная и логарифмическая аппроксимация.
27. Расчет коэффициентов экспоненциальной и логарифмической аппроксимации.

28. Реализация экспоненциальной и логарифмической аппроксимации на языке программирования высокого уровня.

ОПК-7.1.

29. Математический нейрон Маккалока-Питса.
30. Функции активации нейронов.
31. Понятие искусственной нейронной сети.
32. Классификация искусственных нейронных сетей.
33. Персептрон Розенблатта.
34. Однослойные нейронные сети прямого распространения.
35. Многослойные нейронные сети прямого распространения.
36. Обучение нейронных сетей.
37. Алгоритм Розенблатта.
38. Дельта-правило.
39. Алгоритм обратного распространения ошибки.
40. Нейросетевая аппроксимация на основе однослойной сети.
41. Нейросетевая аппроксимация с помощью многослойной сети.
42. Преобразование Фурье.
43. Дискретное преобразование Фурье.
44. Быстрое преобразование Фурье.
45. Одномерное быстрое преобразование Фурье.
46. Алгоритмы одномерного быстрого преобразования Фурье.
47. Реализация быстрого одномерного преобразования Фурье.
48. Двумерное быстрое преобразование Фурье.
49. Алгоритмы двумерного быстрого преобразования Фурье.
50. Реализация быстрого двумерного преобразования Фурье.