

Санкт-Петербургский Государственный Политехнический Университет
Факультет Технической Кибернетики
Кафедра Информационных и Управляющих Систем

ОТЧЕТ
о лабораторной работе № 2

**«Дифференциальные
системы»**
по численному анализу

Работу выполнил студент:

Голубева А. С.
гр. 3084/1

Преподаватель:

Зимницкий В. А.

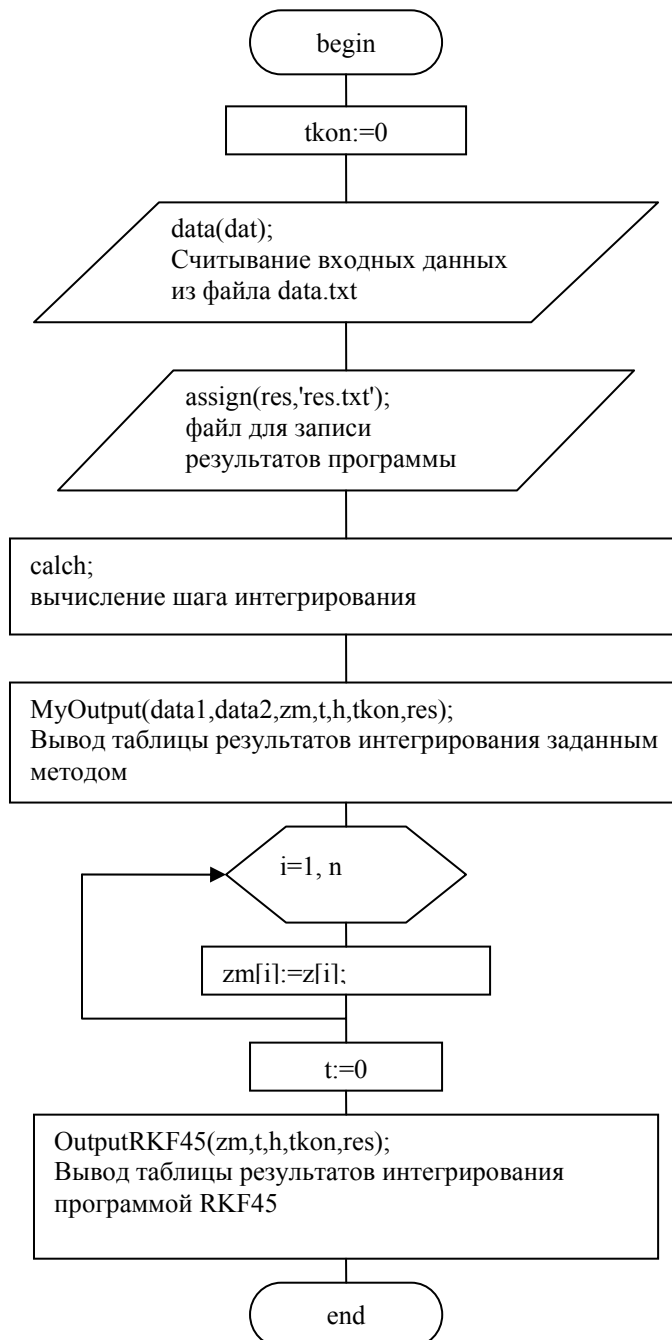
Постановка задачи.

Написать программу интегрирования системы методом:

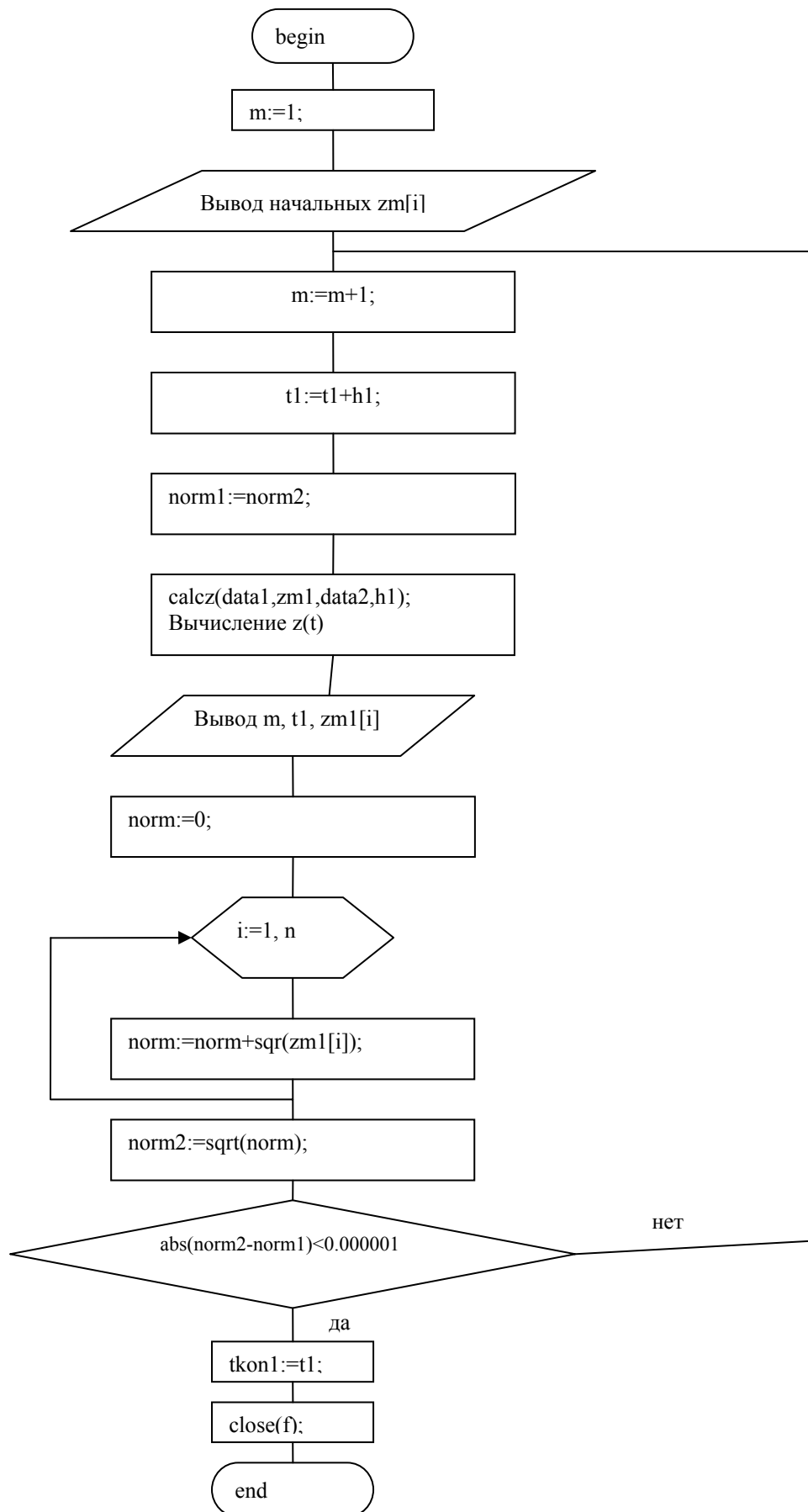
$$z_{n+2/3} = z_n + \frac{2h}{3} f(t_n, z_n)$$
$$z_{n+1} = z_n + \frac{h}{4} \left(f(t_n, z_n) + 3f\left(t_{n+2/3}, z_{n+2/3}\right) \right)$$

Вычислительный эксперимент состоит в анализе точностных характеристик заданного метода по сравнению с решением, которое дает программа RKF45. Шаг для интегрирования заданным методом выбрать по норме $h = \frac{2}{\|A\|}$.

1. Блок-схема алгоритма главной программы.



- Процедура вычисления заданным методом.



2. Исходный текст программы на Паскале (2.pas).

```
program lab2;
uses fmm, crt;
const n=3;
type
    data0=array[1..n,1..n] of real;
var
    t,h,tkon: float;
    data1: data0;
    i,j: byte;
    iflag: integer;
    iwork: ivec5;
    work: rvecn;
    data2,zm,z: floatvector;
    dat,res: text;
{
procedure data(var f: text); {Input data}
begin
    assign(f,
'd: \ds\spri tik\polytech\predmet\math\numana~1\lab\2\data.txt' );
    reset(f);
    for i:=1 to n do
        if i=n then
            begin
                readln(f,zm[i]);
                z[i]:=zm[i];
            end
        else
            begin
                read(f,zm[i]);
                z[i]:=zm[i];
            end;
    for i:=1 to n do
        for j:=1 to n do
            if (i=n)and(j=n) then
                readln(f,data1[i,j])
            else
                read(f,data1[i,j]);
    for i:=1 to n do
    begin
        read(f,data2[i]);
    end;
    close(f);
    end;
}
procedure calch; {Calculation of a step of integration}
begin
    h:=0;
    for i:=1 to n do
    begin
        for j:=1 to n do
        begin
            h:=h+sqr(data1[i,j]);
        end;
    end;
    t:=sqrt(h);
    h:=2/t;
    t:=0;
end;
{
```

```

procedure      prodm(data11: data0; a: fl oatvector; var      cal : fl oatvector);
{Product of matri xes}
var
    i ,j : byte;
begin
    for i :=1 to n do
        for j :=1 to n do
            cal [i ]:=cal [i ]+data11[i , j ]*a[j ];
end;

{_____}
procedure      mul tmn(a: fl oatvector; k: real ; var      cal : fl oatvector);
{Mul ti plication of a matri x to number}
var
    i : byte;
begin
    for i :=1 to n do
        cal [i ]:=a[i ]*k;
end;

{_____}
procedure      addm(a, b: fl oatvector; var      cal : fl oatvector);      {Addi ti on of
matri xes}
var
    i : byte;
begin
    for i :=1 to n do
        cal [i ]:=a[i ]+b[i ];
end;

{_____}
procedure      cal cf(data11: data0; zm1, data21: fl oatvector; var
fm1: fl oatvector);      {Cal cul ati on f(t, z(t)) }
var
    p: fl oatvector;
begin
    for i :=1 to n do
        p[i ]:=0;
        prodm(data11, zm1, p);
        addm(p, data21, fm1);
end;

{_____}
procedure      cal cz(data11: data0; var
zm1: fl oatvector; data21: fl oatvector; h1: fl oat); {Cal cul ati on z(t)}
var
    p, p1, fm, fm23, zm23: fl oatvector;
begin
    p[i ]:=0;
    p[i ]:=0;
    fm[i ]:=0;
    fm23[i ]:=0;
    zm23[i ]:=0;
    cal cf(data11, zm1, data21, fm);
    mul tmn(fm, 2*h1/3, p);
    addm(zm1, p, zm23);
    cal cf(data11, zm23, data21, fm23);
    mul tmn(fm23, 3, p);
    mul tmn(fm, 1, p1);
    addm(p1, p, p);

```

```

mul tmn(p, h1/4, p1);
addm(zm1, p1, zm1);
end;

```

```

{_____}
procedure MyOutput(data11: data0; data21: floatvector; var
zm1: floatvector; t1, h1: float; var tkon1: float; var f: text);
{Output my results}
var
    i, j: byte;
    m: integer;
    norm, norm1, norm2: real;
begin
    m:=1;
    rewrite(f);
    writeln(f, '                The table of my results                ');
    writeln(f, '');
    write(f, m: 2, ' ');
    write(f, ' t=' , t1: 6: 3, ' ');
    for i:=1 to n do
        write(f, zm1[i]: 10: 6, ' ');
    repeat
        m:=m+1;
        t1:=t1+h1;
        norm1:=norm2;
        cal cz(data1, zm1, data2, h1);
        writeln;
        writeln(f, '');
        write(f, m: 2, ' ');
        write(f, ' t=' , t1: 6: 3, ' ');
        for i:=1 to n do
            write(f, zm1[i]: 10: 6, ' ');
        norm:=0;
        for i:=1 to n do
            norm:=norm+sqr(zm1[i]);
        norm2:=sqrt(norm);
    until abs(norm2-norm1)<0.000001;
    tkon1:=t1;
    close(f);
end;

```

```

{_____}
{$F+}
procedure f(t1: float; var zm1, fm: floatvector);
{Input procedure for RKF45}
var
    p: floatvector;
begin
    for i:=1 to n do
        p[i]:=0;
    prodm(data1, zm1, p);
    addm(p, data2, fm);
end;
{$F-}

```

```

{_____}
procedure checki flag(var i flag1: integer; var abserr1, relerr1: float);
{Check I FLAG}
begin

```

```

case i fl ag1 of
  5: abserr1:=1E-9;
  6: begin
      rel err1:=10*rel err1;
      i fl ag1:=2;
    end;
  7: i fl ag1:=2;
end;
end;

{_____}
procedure      OutputRKF45(var      zm1: fl oatvector; t1, h1, tkon1: fl oat; var
ff: text);    {Output RKF45}
var
  t2, tout, rel err, abserr: fl oat;
  i fl ag, i , m: i nteger;
  i work: i vec5;
  work: rvecn;
  zm23: fl oatvector;
begin
  i fl ag:=1;
  t2:=t1;
  tout:=t1;
  rel err:=1E-9;
  abserr:=0;
  rkf45(@f, n, zm1, t1, tout, rel err, abserr, i fl ag, work, i work);
  append(ff);
  wri tel n(ff, ' ');
  wri tel n(ff, ' ');
  wri tel n(ff, '          The table of RKF45 resul ts          ');
  wri tel n(ff, ' ');
  m:=1;
  wri te(ff, m: 2, ' ');
  wri te(ff, ' t=', t1: 6: 3, ' ');
  for i:=1 to n do
    wri te(ff, ' ', zm1[i]: 10: 6);
  wri tel n(ff, ' ', i fl ag);
  t1:=t2;
  repeat
    m:=m+1;
    checki fl ag(i fl ag, abserr, rel err);
    t2:=t1;
    tout:=tout+h1;
    rkf45(@f, n, zm1, t1, tout, rel err, abserr, i fl ag, work, i work);
    checki fl ag(i fl ag, abserr, rel err);
    wri te(ff, m: 2, ' ');
    wri te(ff, ' t=', t1: 6: 3, ' ');
    for i:=1 to n do
      begin
        wri te(ff, ' ', zm1[i]: 10: 6);
      end;
    wri tel n(ff, ' ', i fl ag);
  until t1>=tkon1;
  close(ff);
end;

{_____}
begin {Main program}
  tkon:=0;
  data(dat);

```

```

assign(res, 'd:\ds\spri ti k\pol ytech\predmet\math\numana~1\lab\2\res. txt' )
;
cal ch;
MyOutput(data1, data2, zm, t, h, tkon, res);
for i:=1 to n do
begin
  zm[i ]:=z[i ];
end;
t:=0;
OutputRKF45(zm, t, h, tkon, res);
end.

```

3. Результаты работы программы (res.txt)

- Таблица результатов интегрирования заданным методом.

	t	z1	z2	z3
1	0,000	-2,000000	1,000000	0,000000
2	0,028	-1,368814	0,698065	1,173159
3	0,057	-1,105439	0,674513	1,696378
4	0,085	-1,005859	0,737062	1,916151
5	0,114	-0,975431	0,811319	1,999393
6	0,142	-0,971724	0,873134	2,024472
7	0,170	-0,976628	0,918139	2,026933
8	0,199	-0,983013	0,948657	2,022110
9	0,227	-0,988477	0,968456	2,016104
10	0,256	-0,992514	0,980921	2,010984
11	0,284	-0,995278	0,988599	2,007188
12	0,312	-0,997085	0,993252	2,004571
13	0,341	-0,998229	0,996037	2,002847
14	0,369	-0,998937	0,997687	2,001745
15	0,397	-0,999368	0,998657	2,001057
16	0,426	-0,999628	0,999223	2,000634
17	0,454	-0,999782	0,999552	2,000377
18	0,483	-0,999873	0,999742	2,000223
19	0,511	-0,999926	0,999852	2,000131
20	0,539	-0,999957	0,999915	2,000077
21	0,568	-0,999975	0,999951	2,000045
22	0,596	-0,999986	0,999972	2,000026
23	0,625	-0,999992	0,999984	2,000015
24	0,653	-0,999995	0,999991	2,000009
25	0,681	-0,999997	0,999995	2,000005

- Результаты интегрирования программой RKF45

	t	z1	z2	z3	IFLAG
1	0.000	-2.000000	1.000000	0.000000	2
2	0.028	-1.229210	0.516239	1.379522	2
3	0.057	-0.974528	0.518335	1.898256	2
4	0.085	-0.916501	0.642465	2.060167	2
5	0.092	-0.914959	0.676166	2.074845	5
6	0.142	-0.944577	0.856276	2.073292	2
7	0.170	-0.964133	0.915821	2.051379	2
8	0.199	-0.978214	0.952214	2.032974	2
9	0.227	-0.987307	0.973488	2.020052	2
10	0.256	-0.992820	0.985541	2.011756	2
11	0.284	-0.996025	0.992215	2.006715	2
12	0.312	-0.997835	0.995848	2.003761	2
13	0.341	-0.998835	0.997800	2.002076	2
14	0.369	-0.999379	0.998840	2.001133	2
15	0.397	-0.999671	0.999389	2.000614	2
16	0.426	-0.999826	0.999678	2.000330	2
17	0.454	-0.999908	0.999830	2.000177	2
18	0.483	-0.999952	0.999910	2.000095	2
19	0.511	-0.999975	0.999952	2.000051	2
20	0.539	-0.999986	0.999975	2.000027	2
21	0.568	-0.999993	0.999986	2.000015	2
22	0.596	-0.999996	0.999993	2.000008	2
23	0.625	-0.999998	0.999996	2.000004	2
24	0.653	-0.999999	0.999998	2.000002	2
25	0.681	-0.999999	0.999999	2.000001	2