**uses** graphABC;

**function** F(x:real; n:integer):real;

**Var** f1,f2:real;

**begin**

N:=1;

f1:=sqr(x\*x\*x\*x\*x-27\*x\*x\*x-N\*N\*x+10);

f2:=sqr(sqr(sqr(x+1)));

F:=-3\*sqrt(N)+f1/f2;

**end**;

**function** FP(x:real; n:integer): real;

**Var** f1,f2,fp1,fp2:real;

**begin**

N:=1;

f1:=sqr(x\*x\*x\*x\*x-27\*x\*x\*x-N\*N\*x+10);

f2:=sqr(sqr(sqr(x+1)));

fp1:=2\*(5\*x\*x\*x\*x-81\*x\*x-N\*N)\*sqrt(f1);

fp2:=8\*f2/(x+1);

Fp:=(fp1\*f2-f1\*fp2)/sqr(f2);

**end**;

**var** x,max,min,mx,my,a,b,eps,s1,s2,h,xn:real;

x0,y0,i,j,nr,n:integer;

**label** Konec;

**begin**

a:=0.2;

b:=10;

max:=F(a,n);

min:=F(a,n);

x:=a;

**while** x<=b **do**

**begin**

**if** F(x,n)>max **then** max:=F(x,n);

**if** F(x,n)<min **then** min:=F(x,n);

x:=x+0.01;

**end**;

x0:=50;

y0:=round(max\*windowheight/(max-min))-50;

mx:=(windowwidth-x0-30)/b;

my:=(y0-30)/max;

line(0,y0,windowwidth,y0);

line(x0,0,x0,windowheight);

**for** i:=1 **to** 20 **do**

**begin**

line(x0+round(i\*mx/2),y0-3,x0+round(i\*mx/2),y0+3);

textout(x0+round(i\*mx/2),y0+1,floattostr(i/2));

line(x0-3,y0+round(i\*my),x0+3,y0+round(i\*my));

line(x0-3,y0-round(i\*my),x0+3,y0-round(i\*my));

textout(x0-25,y0-round(i\*my),inttostr(i));

textout(x0-20,y0+round(i\*my),inttostr(-i));

**end**;

textout(x0+5,y0+10,'0');

textout(windowwidth-10,y0-15,'X');

textout(x0+5,10, 'Y');

x:=a;

**while** x<=b **do**

**begin**

setpixel(x0+round(x\*mx),y0-round(F(x,n)\*my),clBlue);

x:=x+0.001;

**end**;

setfontsize(12);

{Метод Ньютона}

eps:=0.001;

textout(80,0,'N=');

readln(n);

textout(100,0,n);

textout(80,20,'Введите первый корень уравнения: ' );

readln(x);

textout(400,20,x);

i:=0;

xn:=0;

**Repeat**

**begin**

xn:=x;

x:=x-F(x,n)/FP(x,n);

i:=i+1;

**if** i>1000 **then**

**begin**

writeln('Точность не достигнута');

readln;

**goto** konec;

**end**;

**end**;

**until** abs(x-xn)<eps;

textout(80,40,'Корень уравнения x=');

textout(250,40,x);

textout(290,40,' ');

textout(80,60,'Введите второй корень уравнения: ' );

readln(x);

textout(400,60,x);

i:=0;

xn:=0;

**Repeat**

**begin**

xn:=x;

x:=x-F(x,n)/FP(x,n);

i:=i+1;

**if** i>1000 **then**

**begin**

writeln('Точность не достигнута');

readln;

**goto** konec;

**end**;

**end**;

**until** abs(x-xn)<eps;

textout(80,80,'Корень уравнения x=');

textout(250,80,x);

textout(290,80,' ');

{ Вычисление интеграла метод Симпсона}

nr:=100; {Число разбиений}

textout(80,100,'Введите нижний предел интегрирования a=');

readln(a);

textout(400,100,a);

textout(80,120,'Введите верхний предел интегрирования b=');

readln(b);

textout(400,120,b);

S1:=0;

h:=(b-a)/nr;

x:=a;

i:=0;

**while** x<b+h/2 **do**

**begin**

**if** ((i=0) **or** (i=NR)) **then**

S1:=S1+F(x,n)

**else**

**begin**

**if** (i **mod** 2>0) **then**

S1:=S1+2\*F(x,n)

**else**

S1:=S1+4\*F(x,n);

**end**;

x:=x+h;

i:=i+1;

**end**;

textout(80,140,'Интеграл (метод Cимпсона):');

textout(300,140,(S1)\*h/3);

textout(361,140,' ');

Konec:

**end**.