

Лабораторная работа № 3

Детерминированные вычислительные процессы с управлением по аргументу. Численное интегрирование

Цель: научиться реализовать алгоритм детерминированных вычислительных процессов с управлением по аргументу средствами Free Pascal

Оборудование: компьютер, PascalABC.NET

Задание 1

Написать программу для вычисления определенного интеграла из индивидуального задания методом прямоугольника левых частей.

Протестировать программу на определенном интеграле, вычисленном в ходе выполнения самостоятельной работы 3

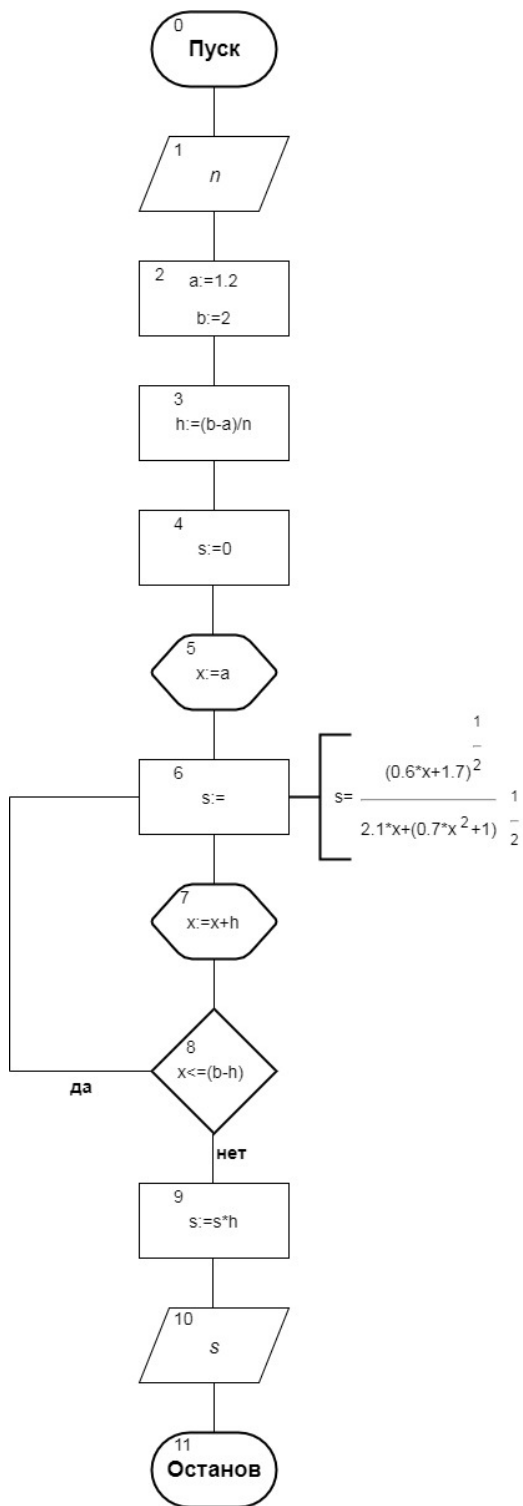
Математическая модель:

$$\int_{1,2}^2 \frac{\sqrt{0,6x+1,7} dx}{2,1x+\sqrt{0,7x^2+1}};$$

$$h = \frac{(b - a)}{n}$$

$$x = x + h$$

Блок схема:



Список идентификаторов:

<i>Имя</i>	<i>Смысл</i>	<i>Тип</i>
<i>s</i>	искомое значение интеграла	real
<i>x</i>	параметр цикла	real
<i>a</i>	нижняя граница интеграла	real
<i>h</i>	шаг	real
<i>n</i>	кол-во разбиений	integer
<i>b</i>	верхняя граница интеграла	integer

Код программы:

```
program zadanie1;

var  s, x, a, h:real;

      n, b:integer;

begin

      writeln('Vvedite n');

      readln(n);

      a:= 1.2;

      b:= 2;

      h:= (b-a)/n;

      s:= 0;

      x:= a;

      while x<= (b-h) do begin

              s:= power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2));

              x:= x+h;

      end;

      s:=s*h;

      writeln('integral raven ',s:4:4);

end.
```

Результаты:

```

•Program1.pas*
program zadanie1;
var    s, x, a, h:real;
        n, b:integer;
begin
  writeln('Vvedite n');
  readln(n);
  a:= 1.2;
  b:= 2;
  h:= (b-a)/n;
  s:= 0;
  x:= a;
  while x<= (b-h) do begin
    s:= power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x+1),1/2));
    x:= x+h;
  end;
  s:=s*h;
  writeln('integral raven ',s:4:4);
end.

```

Окно вывода

```

Vvedite n
50
integral raven 0.0045

```

Анализ результата: результат расчёта значения был получен с помощью запуска цикла «while» с помощью счетчика $x \leq (b-h)$, использования переменных h , нижнего и верхнего пределов интеграла a, b и переменной s для получения значения интеграла. Полученный результат считается с помощью метода прямоугольника левых частей и представлен в типе *real*

Задание 2

Написать программу для вычисления определенного интеграла из индивидуального задания методом прямоугольника правых частей.
Протестировать программу на определенном интеграле, вычисленном в ходе выполнения самостоятельной работы 3

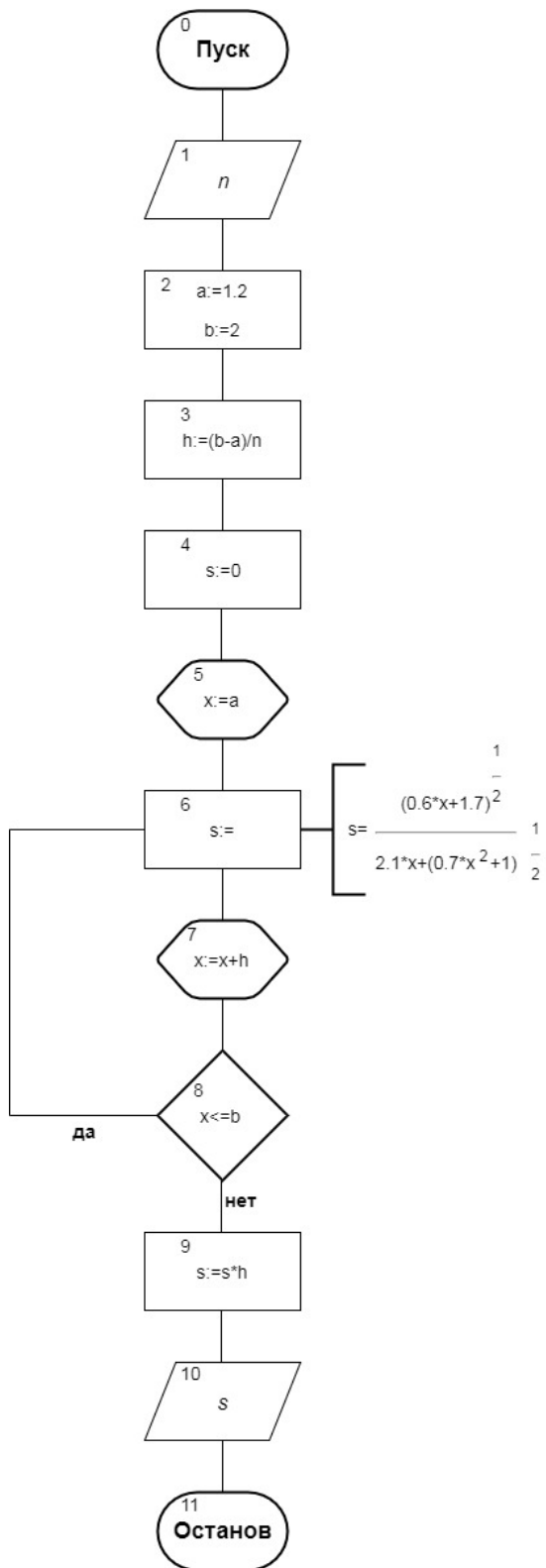
Математическая модель:

$$\int_{1,2}^2 \frac{\sqrt{0,6x+1,7} dx}{2,1x+\sqrt{0,7x^2+1}};$$

$$h = \frac{(b - a)}{n}$$

$$x = x + h$$

Блок схема:



Список идентификаторов:

Имя	Смысл	Тип
<i>s</i>	искмое значение интеграла	real
<i>x</i>	параметр цикла	real
<i>a</i>	нижняя граница интеграла	real
<i>h</i>	шаг	real

n	кол-во разбиений	integer
b	верхняя граница интеграла	integer

Код программы:

```

program zadanie2;

var   s, x, a, h:real;

      n, b:integer;

begin

    writeln('Vvedite n');

    readln(n);

    a:= 1.2;

    b:= 2;

    h:= (b-a)/n;

    s:= 0;

    x:= a+h;

    while x<= b do begin

        s:= power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2));

        x:= x+h;

    end;

    s:=s*h;

    writeln('integral raven ',s:4:4);

end.

```

Результаты:

```

•Program1.pas*
program zadanie2;
var    s, x, a, h:real;
        n, b:integer;
begin
  writeln('Vvedite n');
  readln(n);
  a:= 1.2;
  b:= 2;
  h:= (b-a)/n;
  s:= 0;
  x:= a+h;
  while x<= b do begin
    s:= power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2));
    x:= x+h;
  end;
  s:=s*h;
  writeln('integral raven ',s:4:4);
end.

```

Окно вывода

```

Vvedite n
50
integral raven 0.0045

```

Анализ результата: результат расчёта значения был получен с помощью запуска цикла «while» с помощью счетчика $x \leq b$, использования переменных h , нижнего и верхнего пределов интеграла a, b и переменной s для получения значения интеграла. Полученный результат считается с помощью метода прямоугольника правых частей и представлен в типе *real*

Задание 3

Написать программу для вычисления определенного интеграла из индивидуального задания методом трапеций. Протестировать программу на определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3

Математическая модель:

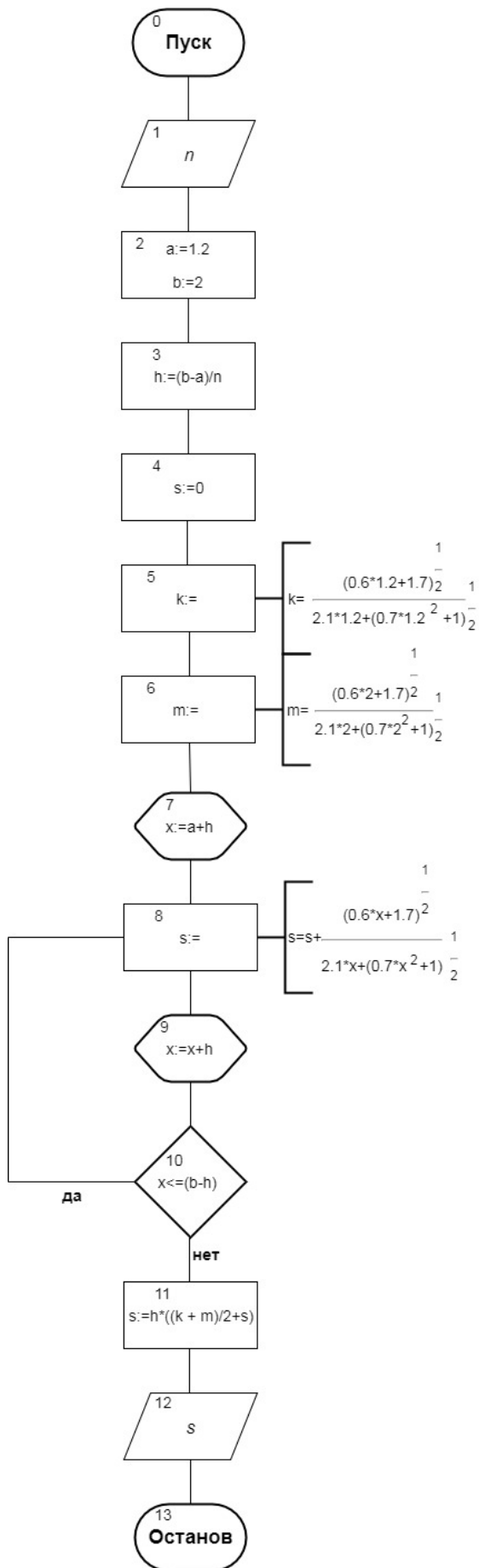
$$\int_{1,2}^2 \frac{\sqrt{0,6x+1,7} dx}{2,1x+\sqrt{0,7x^2+1}};$$

$$h = \frac{(b - a)}{n}$$

$$s = h * \frac{y_1 + y_2}{2} + s$$

$$x = x + h$$

Блок схема:



Список идентификаторов:

Имя	Смысл	Тип
<i>s</i>	искомое значение интеграла	real
<i>x</i>	параметр цикла	real
<i>a</i>	нижняя граница интеграла	real
<i>h</i>	шаг	real
<i>k</i>	функция от a	real
<i>m</i>	функция от b	real
<i>n</i>	кол-во разбиений	integer
<i>b</i>	верхняя граница интеграла	integer

Код программы:

```
program zadanie3;
```

```
var  s, x, a, h, k, m:real;
```

```
      n, b:integer;
```

```
begin
```

```
    writeln('Vvedite n');
```

```
    readln(n);
```

```
    a:= 1.2;
```

```
    b:= 2;
```

```
    h:= (b-a)/n;
```

```
    s:= 0;
```

```
    x:= a+h;
```

```
    k:= power((0.6*1.2+1.7),1/2)/(2.1*1.2+power((0.7*1.2*1.2+1),1/2));
```

```
    m:= power((0.6*2+1.7),1/2)/(2.1*2+power((0.7*2*2+1),1/2));
```

```
    while x<= (b-h) do begin
```

$s := s + \text{power}((0.6 * x + 1.7), 1/2) / (2.1 * x + \text{power}((0.7 * x * x + 1), 1/2));$

$x := x + h;$

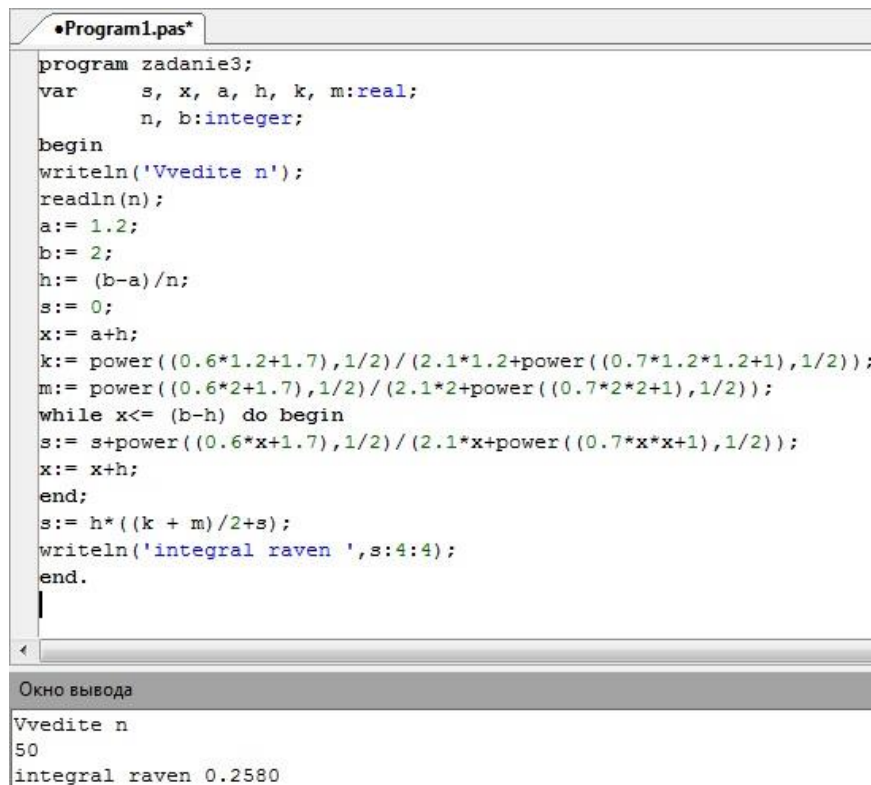
end;

$s := h * ((k + m) / 2 + s);$

writeln('integral raven ', s:4:4);

end.

Результаты:



The screenshot shows a Pascal program editor window titled "Program1.pas" containing the following code:

```
program zadanie3;
var    s, x, a, h, k, m:real;
        n, b:integer;
begin
writeln('Vvedite n');
readln(n);
a:= 1.2;
b:= 2;
h:= (b-a)/n;
s:= 0;
x:= a+h;
k:= power((0.6*1.2+1.7),1/2)/(2.1*1.2+power((0.7*1.2*1.2+1),1/2));
m:= power((0.6*2+1.7),1/2)/(2.1*2+power((0.7*2*2+1),1/2));
while x<= (b-h) do begin
s:= s+power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2));
x:= x+h;
end;
s:= h*((k + m)/2+s);
writeln('integral raven ',s:4:4);
end.
```

Below the editor is a window titled "Окно вывода" (Output Window) showing the program's execution:

```
Vvedite n
50
integral raven 0.2580
```

Анализ результата: результат расчёта значения был получен с помощью запуска цикла «while» с помощью счетчика $x \leq (b-h)$, использования переменных h , нижнего и верхнего пределов интеграла a, b и переменной s для получения значения интеграла. Полученный результат считается с помощью метода трапеций и представлен в типе *real*

Задание 4

Написать программу для вычисления определенного интеграла из индивидуального задания методом парабол. Протестировать программу на

определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3

Математическая модель:

$$\int_{1,2}^2 \frac{\sqrt{0,6x + 1,7} dx}{2,1x + \sqrt{0,7x^2 + 1}};$$

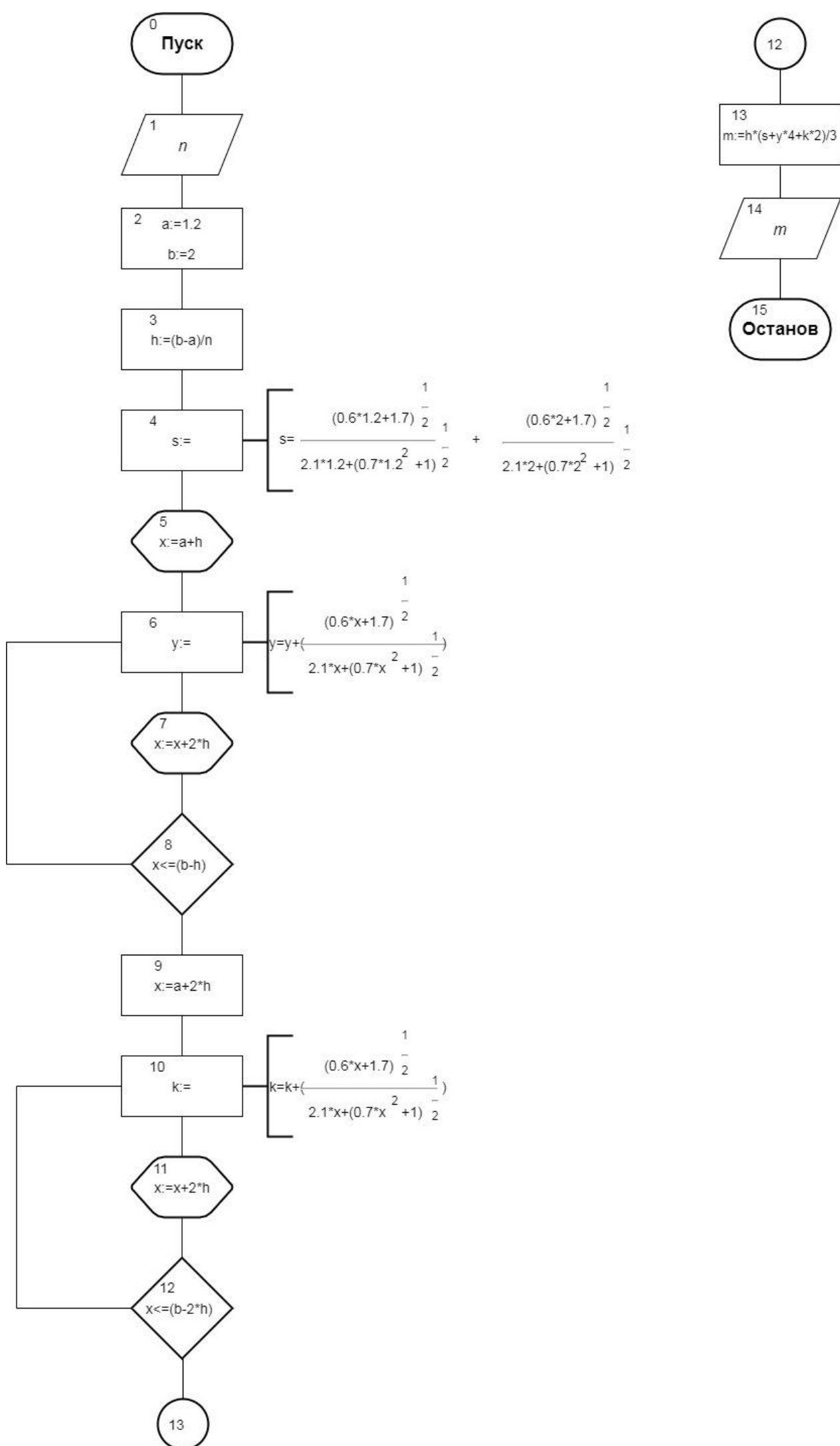
$$s = \frac{\sqrt{0.6 * 1.2 + 1.7}}{2.1 * 1.2 + \sqrt{0.7 * 1.2^2 + 1}} + \frac{\sqrt{0.6 * 2 + 1.7}}{2.1 * 2 + \sqrt{0.7 * 2^2 + 1}}$$

$$y = y + \frac{\sqrt{0.6 * x + 1.7}}{2.1 * x + \sqrt{0.7 * x^2 + 1}}$$

$$k = k + \frac{\sqrt{0.6 * x + 1.7}}{2.1 * x + \sqrt{0.7 * x^2 + 1}}$$

$$m = \frac{h * (s + y * 4 + k * 2)}{3}$$

Блок схема:



Список идентификаторов:

Имя	Смысл	Тип
<i>m</i>	искомое значение интеграла	real
<i>x</i>	параметр цикла	real
<i>a</i>	нижняя граница интеграла	real
<i>h</i>	шаг	real
<i>y</i>	сумма значений интеграла при одинарном шаге	real
<i>k</i>	сумма значений интеграла при двойном шаге	real
<i>s</i>	сумма значений граничных значений интеграла	real
<i>n</i>	кол-во разбиений	integer
<i>b</i>	верхняя граница интеграла	integer

Код программы:

```
program zadanie4;
```

```
var  s, x, a, h, y, k, m:real;
```

```
      n, b:integer;
```

```
begin
```

```
    writeln('Vvedite n');
```

```
    readln(n);
```

```
    a:= 1.2;
```

```
    b:= 2;
```

```
    h:= (b-a)/n;
```

```
    s:= (power((0.6*1.2+1.7),1/2)/(2.1*1.2+power((0.7*1.2*1.2+1),1/2)))+  
        (power((0.6*2+1.7),1/2)/(2.1*2+power((0.7*2*2+1),1/2)));
```

```
    x:= a+h;
```

```
    while x<= (b-h) do begin
```

```
        y:= y+(power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2)));
```

$x := x + 2 * h;$

end;

$x := a + 2 * h;$

while $x \leq (b - 2 * h)$ do begin

$k := k + (\text{power}((0.6 * x + 1.7), 1/2) / (2.1 * x + \text{power}((0.7 * x * x + 1), 1/2)));$

$x := x + 2 * h;$

end;

$m := h * (s + y * 4 + k * 2) / 3;$

writeln('integral raven ', m:4:4);

end.

Результаты:

```
•Program1.pas*
begin
writeln('Vvedite n');
readln(n);
a:= 1.2;
b:= 2;
h:= (b-a)/n;
s:= (power((0.6*1.2+1.7),1/2)/(2.1*1.2+power((0.7*1.2*1.2+1),1/2)))+(power((0.6*2+1.7),1/2)/(2.1*2+power((0.7*2*2+1),1/2)));
x:= a+h;
while x<= (b-h) do begin
y:= y+(power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2)));
x:= x+2*h;
end;
x:=a+2*h;
while x<=(b-2*h) do begin
k:= k+(power((0.6*x+1.7),1/2)/(2.1*x+power((0.7*x*x+1),1/2)));
x:=x+2*h;
end;
m:=h*(s+y*4+k*2)/3;
writeln('integral raven ',m:4:4);
end.
|
```

Окно вывода

```
Vvedite n
50
integral raven 0.2535
```

Анализ результата: результат расчёта значения был получен с помощью запуска цикла «while» с помощью счетчиков $x \leq (b - h)$ и $x \leq (b - 2 * h)$, использования переменных h , нижнего и верхнего пределов интеграла a , b и переменных s, y, k для получения значения интеграла. Полученный результат считается с помощью метода парабол и представлен в типе *real*

Таблица:

Количество разбиений	Шаг	Метод левых частей прямоугольников	Метод правых частей прямоугольников	Метод трапеций	Метод парабол
10	0.0800	0.0235	0.0228	0.2397	0.2625
100	0.0080	0.0022	0.0022	0.2602	0.2580
1000	0.0008	0.0002	0.0002	0.2625	0.2620
10000	0.0001	0.0000	0.0000	0.2624	0.2625

Вывод:

1. Наиболее точным методом является «метод парабол», т.к именно при это методе погрешность наиболее минимальна
2. Точность любого метода можно увеличить, если увеличить отображение цифр после запятой в шаге (h) и искомого значения интеграла (s, m)