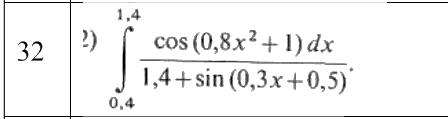
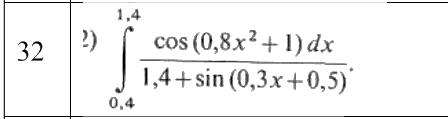
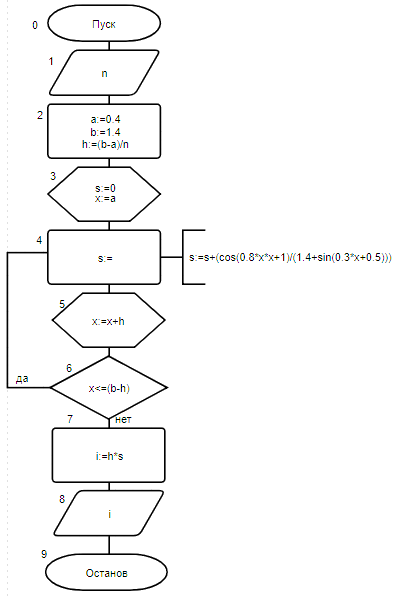
Лабораторная работа №3.  
  
**Тема:** Детерминированные вычислительные процессы с управлением по аргументу. Численное интегрирование.  
  
**Цель**: Реализовать вычисление определенного интеграла при помощи детерминированных вычислительных процессов и различных мметодов нахождения интеграла.   
  
  
**Оборудование:** ПК, материалы лекций, компилятор Pascal ABC.  
  
Задание №1.  
  
1. **Постановка задачи**: Написать программу для вычисления определенного интеграла из индивидуального задания методом прямоугольника левых частей. Протестировать программу на определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3.  
  
2. **Математическая модель:**



3. **Блок-схема:**   
  
  
4. **Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Переменная** | **Смысл** | **Тип** |
| a,b | Пределы интегрирования | real |
| n | Количество разбиений | Integer |
| h | Шаг | real |
| x | Счетчик цикла | real |
| s | Сумма цикла | real |
| i | Выходные данные, результат | real |

5. **Код программы**:   
**program** z1;

**var** a, b, i, s, x, h: real;

n: integer;

**begin**

readln(n);  
a:=0.4;

b:=1.4;

h:=(b-a)/n;

s:=0;

x:=a;

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

s:=s+(cos(0.8\*x\*x+1)/(1.4+sin(0.3\*x+0.5)));

x:=x+h;

**end**;

i:=h\*s;

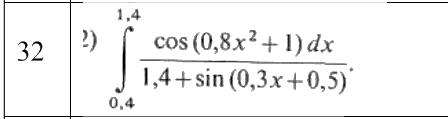
writeln('результат при n=',n,': ', i);

**end**.

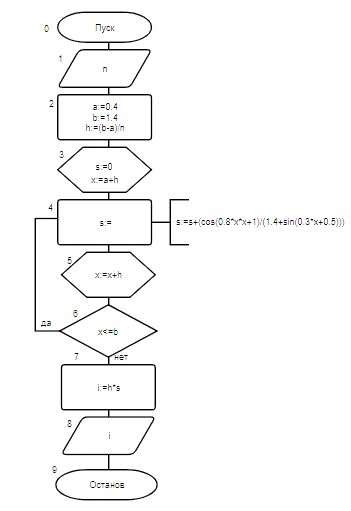
6. **Результат выполненной работы:**




Задание №2.  
  
1. **Постановка задачи**: Написать программу для вычисления определенного интеграла из индивидуального задания методом прямоугольника правых частей. Протестировать программу на определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3.  
  
2. **Математическая модель:**



3. **Блок-схема**:   
  
  
4. **Список идентификаторов**:

|  |  |  |
| --- | --- | --- |
| **Переменная** | **Смысл** | **Тип** |
| a,b | Пределы интегрирования | real |
| n | Количество разбиений | Integer |
| h | Шаг | real |
| x | Счетчик цикла | real |
| s | Сумма цикла | real |
| i | Выходные данные | real |

5. **Код программы:**   
**program** z2;

**var** a, b, i, s, x, h: real;

n: integer;

**begin**

readln(n);

a:=0.4;b:=1.4;

h:=(b-a)/n;

s:=0;

x:=a+h;

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

s:=s+(cos(0.8\*x\*x+1)/(1.4+sin(0.3\*x+0.5)));

x:=x+h;

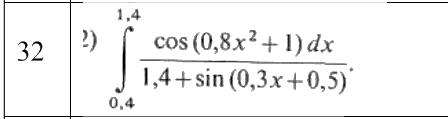
**end**;

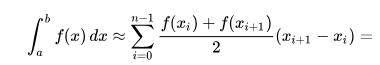
i:=h\*s;

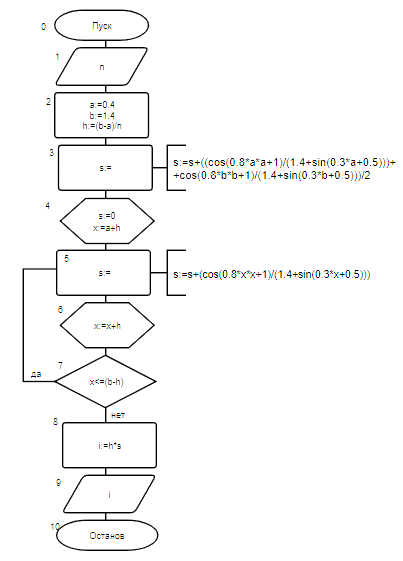
writeln('результат при n=',n,': ', i);

**end**.

6. **Результат выполненной работы:**

  
  
  
  
  
Задание №3.  
  
1. **Постановка задачи**: Написать программу для вычисления определенного интеграла из индивидуального задания методом трапеций. Протестировать программу на определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3.  
  
2. **Математическая модель:**

3. **Блок-схема:**   
  
4. **Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Переменная** | **Смысл** | **Тип** |
| a,b | Пределы интегрирования | real |
| n | Количество разбиений | Integer |
| h | Шаг | real |
| x | Счетчик цикла | real |
| s | Сумма цикла | real |
| i | Выходные данные, результат | real |

5. **Код программы:**   
**program** z3;

**var** a, b, i, s, x, h: real;

n: integer;

**begin**

readln(n);

a:=0.4; b:=1.4;

h:=(b-a)/n;

s:=0;

s:=s+((cos(0.8\*a\*a+1)/(1.4+sin(0.3\*a+0.5)))+cos(0.8\*b\*b+1)/(1.4+sin(0.3\*b+0.5)))/2;

x:=a+h;

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

s:=s+(cos(0.8\*x\*x+1)/(1.4+sin(0.3\*x+0.5)));

x:=x+h;

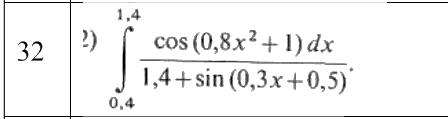
**end**;

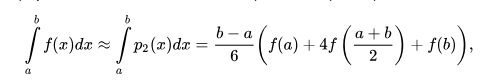
i:=h\*s;

writeln('результат при n=',n,': ', i);

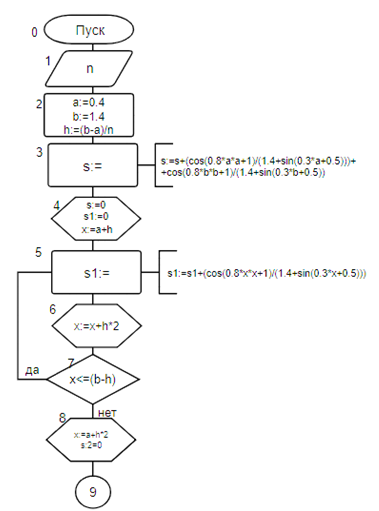
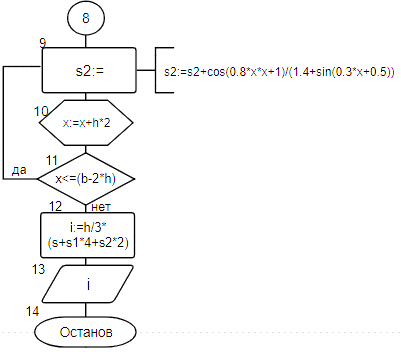
**end**.

6. **Результат выполненной работы:**

  
  
  
  
Задание №4.  
  
1. **Постановка задачи**: Написать программу для вычисления определенного интеграла из индивидуального задания методом парабол. Протестировать программу на определенном интеграле, вычисленным в ходе выполнения самостоятельной работы 3.  
  
2. **Математическая модель:**



3. **Блок-схема:**



4. **Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Переменная** | **Смысл** | **Тип** |
| a,b | Пределы интегрирования | real |
| n | Количество разбиений | Integer |
| h | Шаг | real |
| x | Счетчик цикла | real |
| s, s1, s2 | Суммы цикла | real |
| i | Выходные данные, результат | real |

5. **Код программы:**   
**program** z4;

**var** a, b, i, s, s1, s2, x, h: real;

n: integer;

**begin**

readln(n);

a:=0.4; b:=1.4;

h:=(b-a)/n;

s:=0; s1:=0;

s:=s+(cos(0.8\*a\*a+1)/(1.4+sin(0.3\*a+0.5)))+cos(0.8\*b\*b+1)/(1.4+sin(0.3\*b+0.5));

x:=a+h;

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

s1:=s1+(cos(0.8\*x\*x+1)/(1.4+sin(0.3\*x+0.5)));

x:=x+h\*2;

**end**;

x:=a+2\*h;

s2:=0;

**while** x<=(b-2\*h) **do begin**

s2:=s2+cos(0.8\*x\*x+1)/(1.4+sin(0.3\*x+0.5));

x:=x+2\*h;

**end**;

i:=h/3\*(s+s1\*4+s2\*2);

writeln('результат при n=',n,': ', i);

**end**.

6. **Результат выполненной работы:**


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **N** | **H** | **I метод левый частей прямоугольников** | **I метод правых частей прямоугольников** | **I метод трапеций** | **I метод парабол** |
| 10 | 0.1 | 0.00700649656264571 | -0.0469651802403271 | -0.0229359133652227 | -0.055104886344911 |
| 100 | 0.01 | -0.0483434808843954 | -0.0542789786792332 | -0.0513377218771822 | -0.0475997219312067 |
| 1000 | 0.001 | -0.0548092695458716 | -0.055408117744429 | -0.0551086936451503 | -0.0543449449618587 |
| 10000 | 0.0001 | -0.0550787344550567 | -0.0551386192749124 | -0.0551086768649845 | -0.0551086766954883 |

Вывод:

Где:

N – Количество разбиений;  
H – шаг;  
I – результат интегрирования.

3.1. Наиболее точным является метод парабол, т.к при изменении количества разбиений конечный результат изменяется с минимальной погрешностью, чего нельзя сказать, например, о методе левых частей, где при n=10 результат положительный, а при n=100 отрицательный.   
  
3.2. Точность любого метода можно увеличить при увеличении количества разбиений, т.к остается меньше «непокрытых» зон. При любом методе максимально точный и примерно равный результат получается при n=10000, что является максимальным количеством разбиений в рассмотренных задачах.