**Лабораторная работа №4**

**Детерминированные вычислительные процессы с**

**управлением по аргументу. Численное интегрирование**

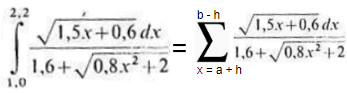
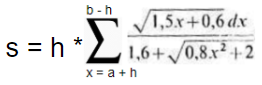
**с использованием функции пользователя.**

**Цель:** научиться [с использованием пользовательской функции](https://moodle.herzen.spb.ru/mod/url/view.php?id=65650) численному интегрированию с помощью алгоритмов детерминированных циклических вычислительных процессов с управлением по аргументу средствами компилятора free Pascal.

**Используемое оборудование:** пк, PascalABC.

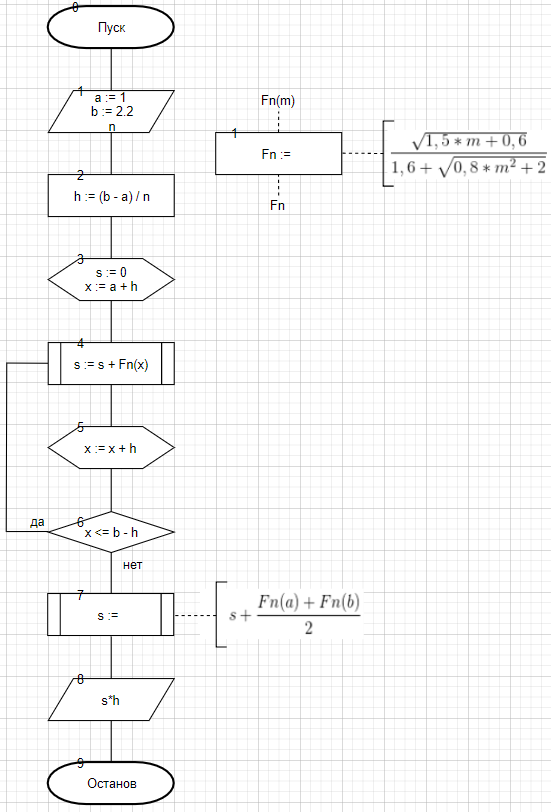
**Задание 1:** метод трапеций.

**4.1)** Реализовать вычисление определенного интеграла из индивидуального задания методом трапеций с использованием пользовательской функции.

**5.1)** ,,,



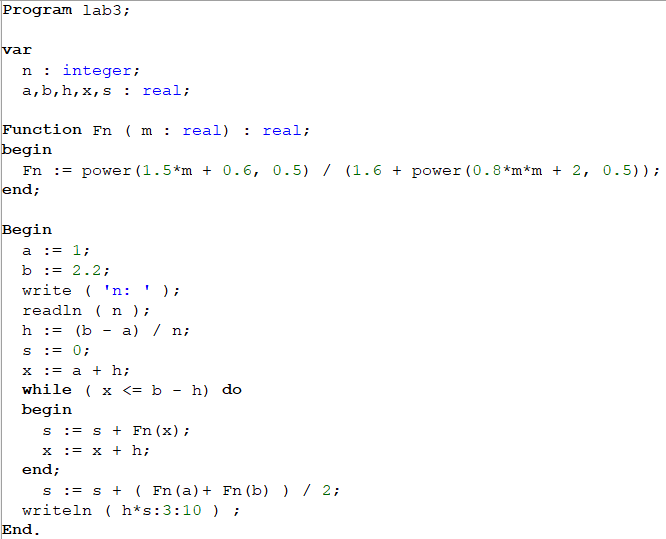
**6.1)**



**7.1)**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| n | Переменная, на сколько частей разбит отрезок [a;b] | Int |
| a | Переменная, левая граница криволинейной функции | Real |
| b | Переменная, правая граница криволинейной функции | Real |
| h | Переменная, шаг | Real |
| x | Переменная, параметр цикла | Real |
| s | Переменная, площадь | Real |
| m | Первый параметр функции Fn(m). | Real |
| Fn( m) | Функция с параметром m для вычисления выражения. | Real |

**8.1)**



**9.1)**

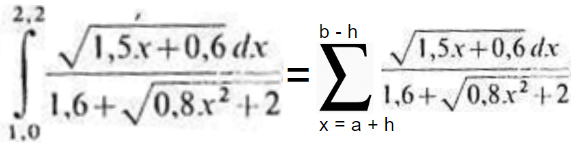
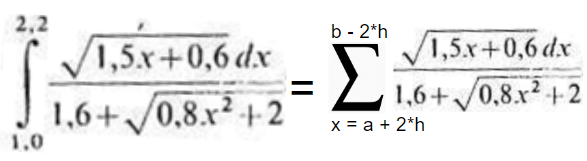
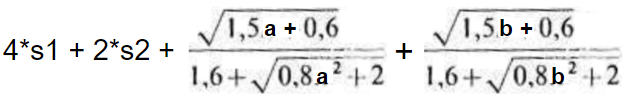
|  |  |  |
| --- | --- | --- |
| **n**  **Количество**  **разбиений** | **h**  **Шаг** | **I**  **Метод**  **трапеций** |
| **10** | **0,12** |  |
| **100** | **0,012** |  |
| **1000** | **0,0012** |  |
| **10000** | **0,00012** |  |

**10.1)** Для решения поставленной задачи мы воспользовались основными математическими операциями, а также для возведения в степень потребовалась функция power. Для удобства мы заменили подынтегральное выражение функцией Fn(m), где m имеет тип real, как и сама функция. Переменная “x” типа real - параметр цикла, где его начальное значение равняется a + h. Шаг “h” типа int выражается (b – a) / n. Переменная “s” типа real содержит вычисленное выражение для последующего вывода на экран. При увеличении количества дроблений “n” отрезка [a, b] точность результатов возрастала.

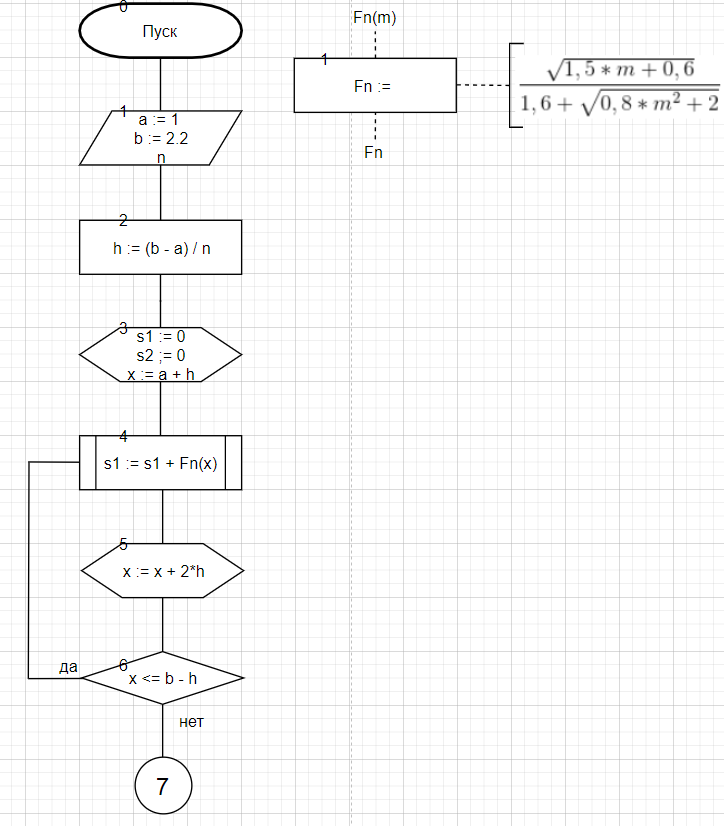
**Задание 2:** метод парабол.

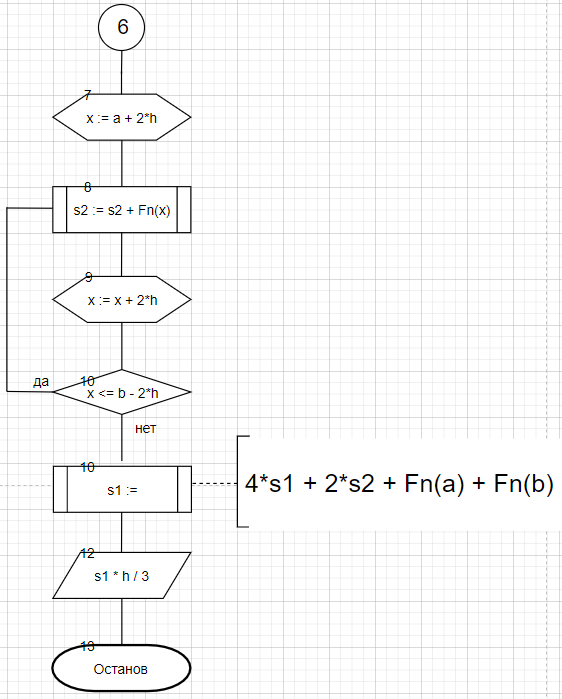
**4.2)** Реализовать вычисление определенного интеграла из индивидуального задания методом парабол с использованием пользовательской функции.

**5.2)**

,,, ,

**6.2)**

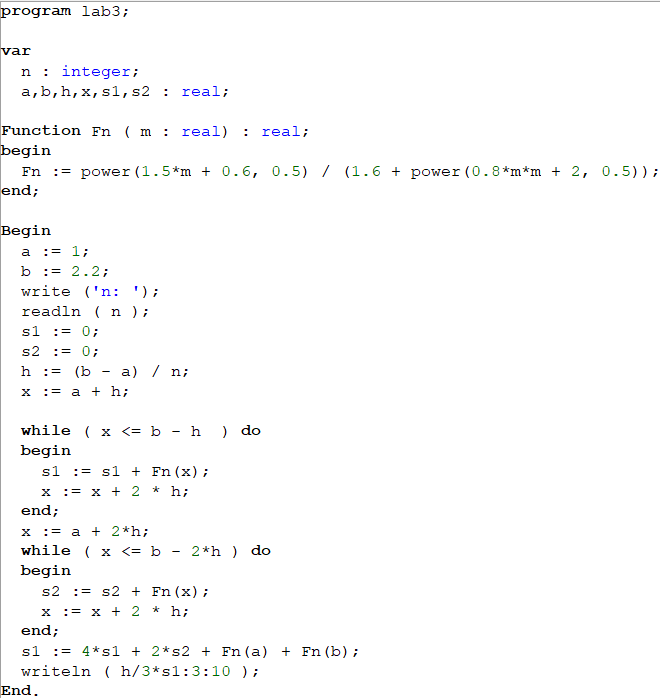




**7.2)**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| n | Переменная, на сколько частей разбит отрезок [a;b] | Int |
| a | Переменная, левая граница криволинейной функции | Real |
| b | Переменная, правая граница криволинейной функции | Real |
| h | Переменная, шаг | Real |
| x | Переменная, параметр цикла | Real |
| s1 | Переменная, сумма некой части, а после итоговая сумма | Real |
| s2 | Переменная, сумма некой части | Real |
| m | Первая переменная функции Fn(m). | Real |
| Fn( m) | Функция с параметром m для вычисления выражения. | Real |

**8.2)**

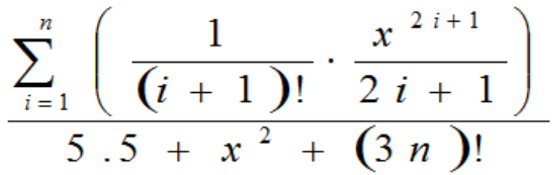


**9.2)**

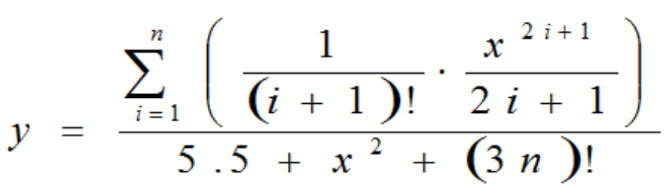
|  |  |  |
| --- | --- | --- |
| **n**  **Количество**  **разбиений** | **h**  **Шаг** | **I**  **Метод парабол** |
| **10** | **0,12** |  |
| **100** | **0,012** |  |
| **1000** | **0,0012** |  |
| **10000** | **0,00012** |  |

**10.2)** Для решения поставленной задачи мы воспользовались основными математическими операциями, а также для возведения в степень потребовалась функция power. Для удобства мы заменили подынтегральное выражение функцией Fn(m), где параметр “m” имеет тип real, как и сама функция. Чтобы реализовать алгоритм, понадобилось два цикла с разными условиями, так для первого цикла условие остановки при x <= b – h, а у второго при x <= b – 2\*h. Переменная “x” типа real - параметр цикла, где его начальное значение равняется a + h для первого цикла, для второго же a + 2\*h. Шаг “h” типа int выражается (b – a) / n. Переменная “s” типа real содержит вычисленное выражение для последующего вывода на экран. Можно заметить, что точность результатов не всегда возрастает при увеличении количества дроблений отрезка [a, b].

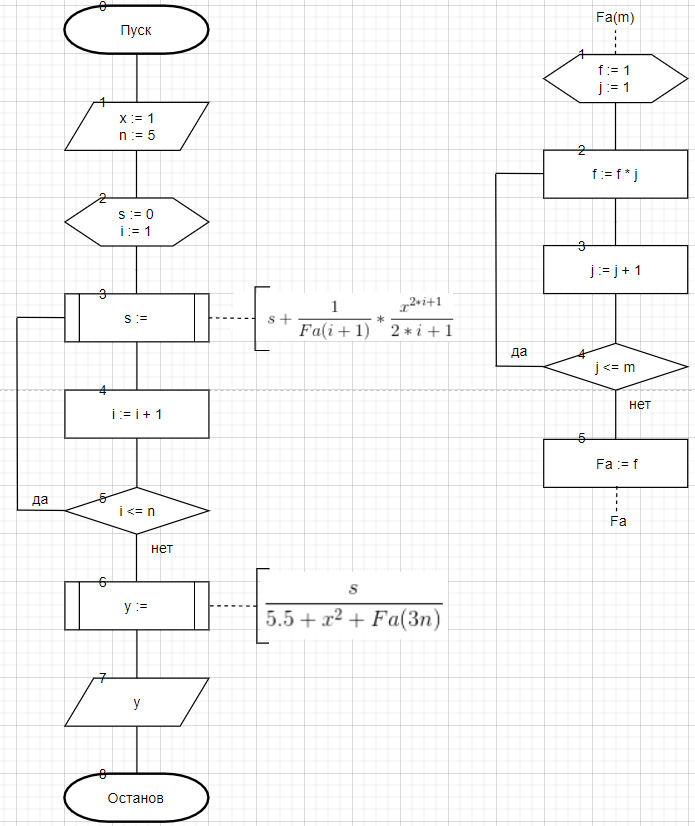
**Задание 3.**

**4.3)** Вычислить результат переменной **y** по выражению, ,

при x = 1, n = 5.

**5.3)** ,

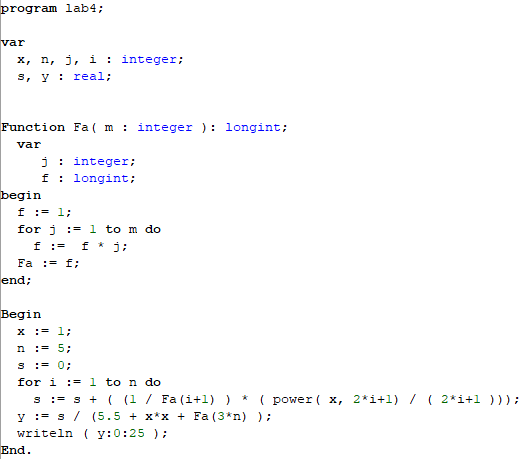
**6.3)**



**7.3)**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| x | Переменная, данное число | int |
| n | Переменная, данный факториал | int |
| j | Переменная функции Fa(m), параметр цикла | int |
| f | Переменная функции Fa(m), факториал | longint |
| i | Переменная, параметр цикла | int |
| s | Переменная, сумма | real |
| y | Переменная, вычисленное значение выражения | real |
| m | Параметр функции, принимающий факториал | int |
| Fa(m) | Функция с параметром m для вычисления факториала | longint |

**8.3)**



\*Как высчитывать факториал сразу через Fa, не придумал, только если через рекурсию, но не стал её использовать.

**9.3)**



**10.3)** Для решения поставленной задачи мы воспользовались основными математическими операциями, а также использовалась функция “power()” . Для удобства мы ввели функцию “Fa(m)”, вычисляющую факториал n, где параметр “n” имеет тип int, а сама функция longint, чтобы вычисляемый факториал не выходил за пределы. Функция содержит в себе переменные j типа int и f типа longint, также чтобы обезопаситься от выхода за пределы факториала. Переменная “s” типа real сумма, которая находиться в числителе выражения. Итоговое значение выражения находиться в переменной “y” типа real. Для получения результата была установлена точность в 25 знаков после запятой.

**11)** Мы научились [с использованием пользовательской функции](https://moodle.herzen.spb.ru/mod/url/view.php?id=65650) численному интегрированию с помощью алгоритмов детерминированных циклических вычислительных процессов с управлением по аргументу средствами компилятора free Pascal.

Для решения поставленной задачи мы воспользовались основными