**Лабораторная работа №8**

**Итерационные циклические вычислительные процессы с управлением по индексу/аргументу и функции.**

**Цель:** научиться итерационным циклическим процессам с управлением по индексу/аргументу и функции средствами компилятора free Pascal.

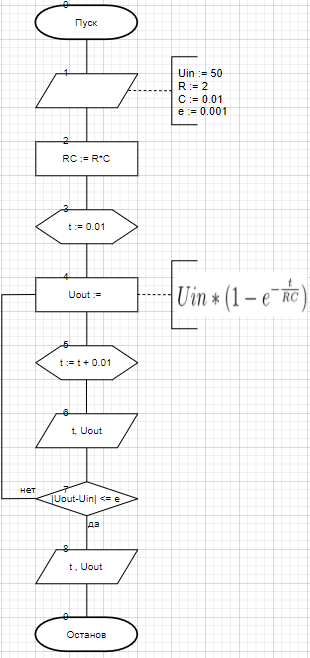
**Используемое оборудование:** пк, PascalABC.

**Задание 1:**

**4.1)** Построить переходную характеристику заряда конденсатора по схеме RC цепочки с заданной точностью e.

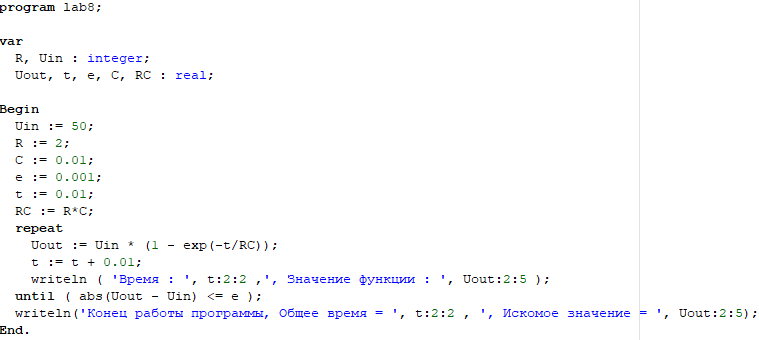
**5.1)** , 

**6.1)**

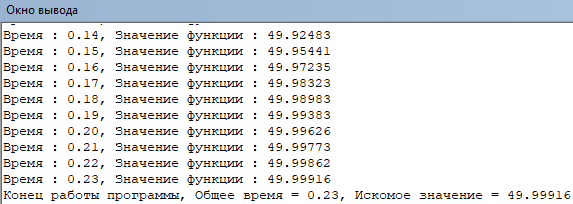


**7.1)**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| **R** | **Переменная, сопротивление.** | **Int.** |
| **Uin** | **Переменная, входное напряжение.** | **Int.** |
| **Uout** | **Переменная, выходное напряжение.** | **Real.** |
| **t** | **Переменная, время.** | **Real.** |
| **e** | **Переменная, заданная точность.** | **Real.** |
| **C** | **Переменная, емкость конденсатора.** | **Real.** |
| **RC** | **Промежуточная переменная, произведение R и С.** | **Real.** |

**8.1)** 

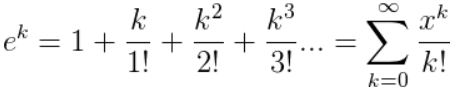
**9.1)**



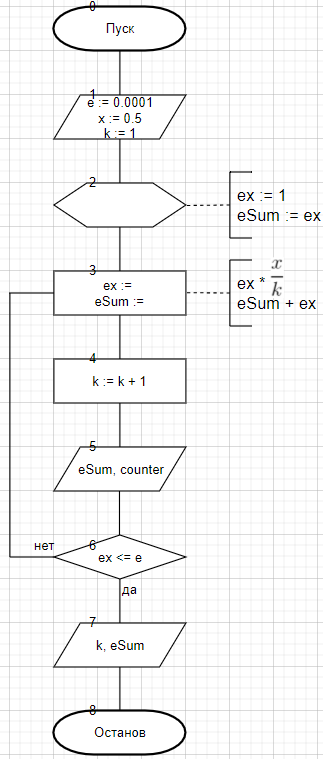
**10.1**) Задача решена при помощи основных математических операций, а также цикла с постусловием repeat until. Сначала работы программы инициализируются переменные, а затем мы попадаем в цикл. В нем первой операцией вычисляется выходное напряжение ”Uout”, после чего время “t” увеличивается на 0.01. Дальше выводится время и значение “Uout” с точностью в 5 знаков после запятой. При истинности условия |Uout-Uin|<= e осуществляется выход из цикла, иначе цикл продолжает работу. Когда цикл завершает работу, выводиться конечное время и искомое значение “Uout” с точностью в пять знаков после запятой.

**Задание 2:**

**4.2)** Вычислить e(x) с точностью 10^(-4). Начальные условия: k=1, x = 0.5, eSum = 1, ex = 1.

**5.2)** , 

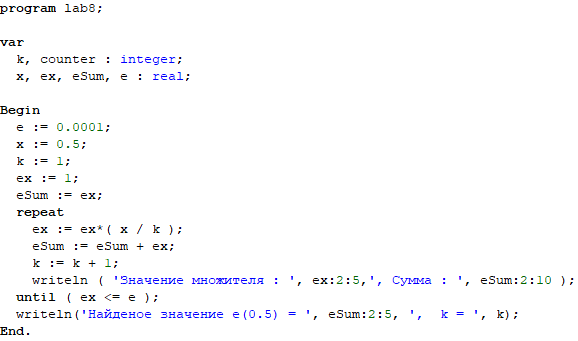
**6.2)**



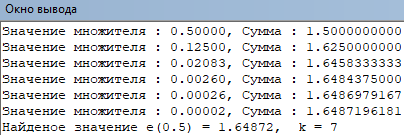
**7.2)**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| **k** | **Переменная, член ряда** | **Int.** |
| **e** | **Переменная, заданная точность.** | **Real.** |
| **x** | **Переменная, аргумент** | **Real.** |
| **eSum** | **Переменная, сумма всех элементов ряда** | **Real.** |
| **ex** | **Переменная, элемент ряда экспоненты от x** | **Real.** |

**8.2)**



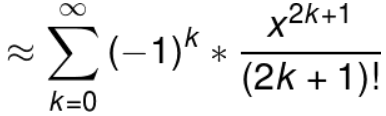
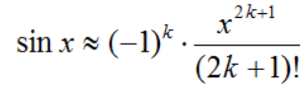
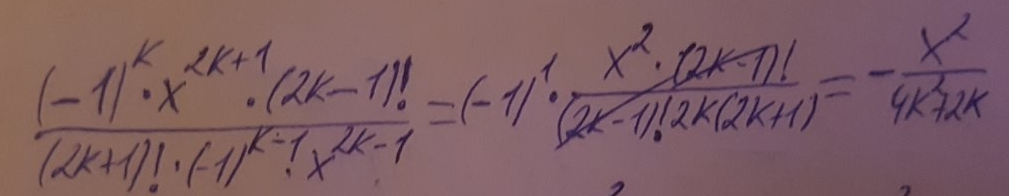
**9.2)**



**10.2)** Задача была выполнена при помощи основных математических операций и цикла с постусловием until repeat. При запуске программы сначала инициализируются переменные, где же и вычисляется первый элемент ряда экспоненты - ex := x^k/k! при k = 0. Следующим шагом мы попадаем в цикл, в котором высчитывается сумма элементов экспоненты от x, каждый элемент находится по формуле x/k, это рекуррентная формула используется для упрощения вычислений компьютеру, вывести её можно, поделив некий элемент ряда на предыдущий ему. Переменная “eSum” содержит сумму элементов ряда, а дальше переменная “k” увеличивается на 1. Следующая операция сумму элементов ”eSum”, затем проверяется условие ex <= e, при истинности которого цикл завершается, в ином случае продолжается, после завершения работы цикла выводится значение переменной “k” и найденная экспонента от x с точностью в пять знаков после запятой.

**Задание 3:**

**4.3)** Вычислить Sin(x) с точностью 10^(-4). Начальные условия: k = 1, uSin = x, sum = x, x = pi/6

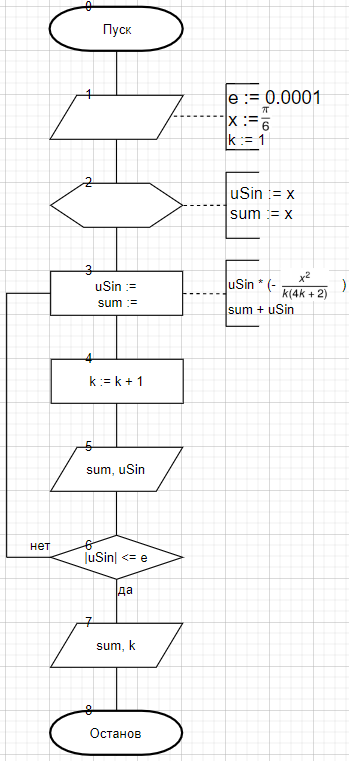
**5.3)** // 

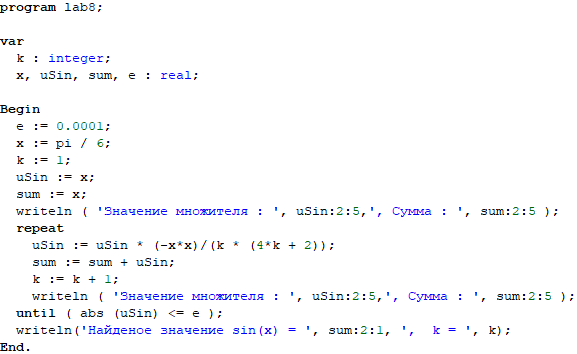
**6.3)**

**7.3)**

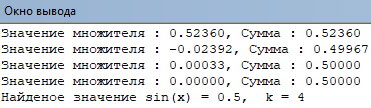
|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| **k** | **Переменная, член ряда.** | **Int.** |
| **x** | **Переменная, угол в радианах.** | **Real.** |
| **uSin** | **Переменная, элемент ряда.** | **Real.** |
| **sum** | **Переменная, сумма, а впоследствии значение синуса от x.** | **Real.** |
| **e** | **Переменная, заданная точность.** | **Real.** |

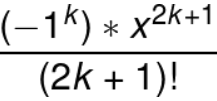
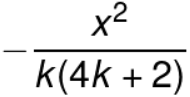
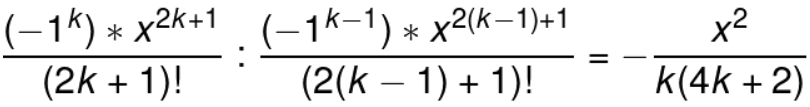
**8.3)**





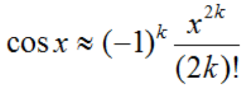
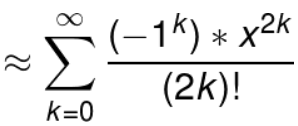
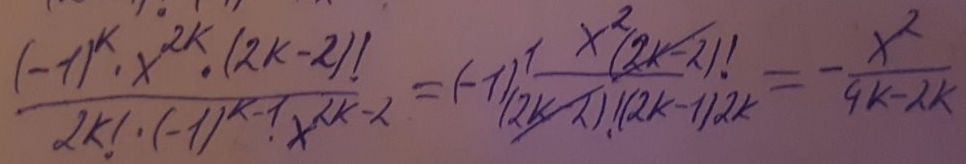
**9.3)**



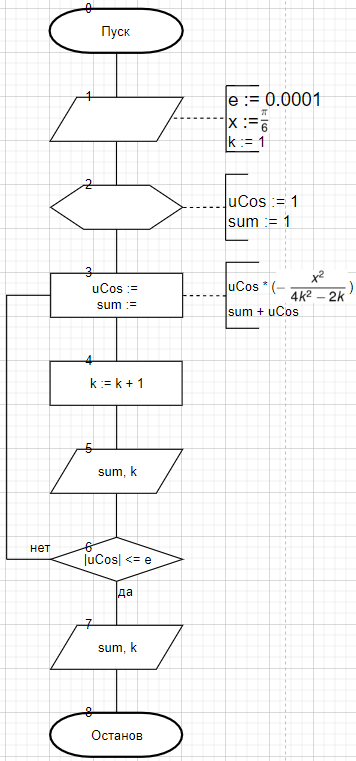
**10.3)** Задача решена при помощи основных математических операций и цикла с постусловием until repeat. При запуске программы сначала инициализируются переменные, где же и вычисляется первый элемент ряда синуса - uSin := при k = 0. Следующим шагом мы попадаем в цикл, в котором высчитывается сумма элементов синуса от x, каждый элемент находится по формуле , это рекуррентная формула используется для упрощения вычислений компьютеру, вывести её можно, поделив некий элемент ряда на предыдущий ему -  . Переменная “sum” содержит сумму элементов ряда, а параметр “k” увеличивается на 1. Следующая сумму элементов ”sum”, затем проверяется условие |uSin|<= e, при истинности которого цикл завершается, в ином случае продолжается, после завершения работы цикла выводится значение переменной “k” и найденный синус от угла x с точностью в один знак после запятой.

**Задание 4:**

**4.4)** Вычислить cos(x) с точностью 10^(-4). Начальные условия: k = 1, uSin = x, sum = x, x = pi/6

**5.4)**  // 

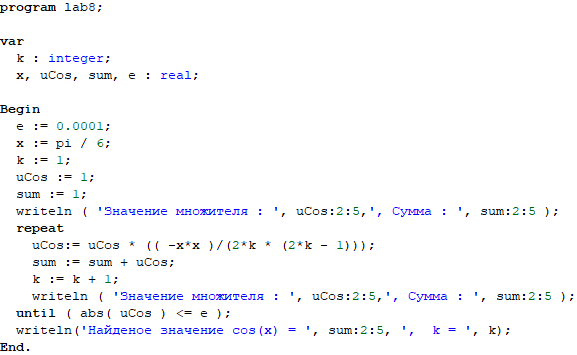
**6.4)**



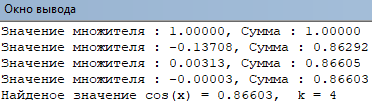
**7.4)**

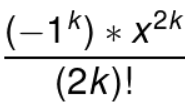
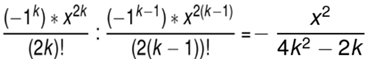
|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| **k** | **Переменная, член ряда.** | **Int.** |
| **x** | **Переменная, угол в радианах.** | **Real.** |
| **uCos** | **Переменная, элемент ряда.** | **Real.** |
| **sum** | **Переменная, сумма, а впоследствии значение косинуса от x.** | **Real.** |
| **e** | **Переменная, заданная точность.** | **Real.** |

**8.4)**



**9.4)**



**10.4)** Задача решена при помощи основных математических операций и цикла с постусловием until repeat. При запуске программы сначала инициализируются переменные, где же и вычисляется первый элемент ряда синуса - uCos :=  при k = 0. Следующим шагом мы попадаем в цикл, в котором высчитывается сумма элементов синуса от x, каждый элемент находится по формуле , это рекуррентная формула используется для упрощения вычислений компьютеру, вывести её можно, поделив некий элемент ряда на предыдущий ему -  . Переменная “sum” содержит сумму элементов ряда, а дальше параметр “k” увеличивается на 1. Следующая операция выводит сумму элементов ”sum”, затем проверяется условие |uCos|<= e, при истинности которого цикл завершается, в ином случае продолжается, после завершения работы цикла выводится значение переменной “k” и найденный косинус от угла x с точностью в пять знаков после запятой.

**11)** Мы научились реализовывать алгоритмы итерационных циклических процессов с управлением по индексу/аргументу и функции средствами компилятора free Pascal.