1. Лабораторная работа № 3

«Программирование алгоритмов циклической структуры»

* 1. Цель работы

Получить навыки программирования итерационных циклических алгоритмов, исследовать зависимость объёма вычислений от точности.

* 1. Постановка задачи

Научиться разрабатывать программы с применением итерационных циклических структур.

* 1. Ход выполнения работы
     1. Математическое обоснование

Требуется определить область определения функции (Вариант 19) (Формула 3.1).

Так как неизвестная находится в числителе дроби, то неизвестная может принимать любое значение. Значит, с учётом поставленного условия (), область определения будет равна:

Также стоит учесть, что при сумма ряда равна нулю, так как каждый член яда будет равен нулю.

Так как каждый член ряда меньше предыдущего, а самый первый находится в промежутке , то – бесконечно убывающий ряд, сумма членов которого стремится к 1, если , и стремится к -1, если . Таким образом область значений будет равна:

* + 1. Структурная схема алгоритма

Структурная схема рабочего алгоритма представлена на Рисунке 3.1.

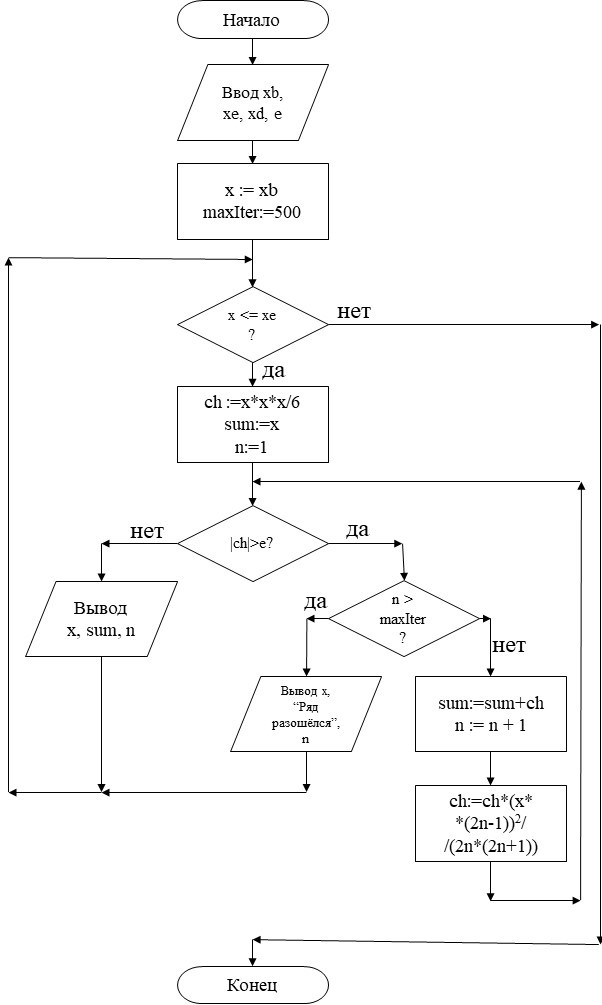


Рисунок 3.1 – Структурная схема алгоритма

* + 1. Написание кода алгоритма

С клавиатуры вводятся значения переменных xb (начальное значение), xe (конечное значение), xd (шаг x), e (точность вычислений). Результат выводится в виде таблицы с заголовком и шапкой.

Program lw3;

Const maxIter = 500;

Var x, xb, xe, xd, e, ch, sum:double;

n: integer;

done: boolean;

{

Список используемых переменных:

xb - начальное значение.

xe - конечное значение.

xd - изменение переменной.

e - заданная точность вычисления.

x - данное значение X.

sum - сумма всех чисел ряда для данного X.

ch - значение текущего члена ряда.

n - номер текущего члена ряда.

done - Истина, если ряд не разошёлся, иначе Ложь.

Список используемых констант:

maxIter - максимальное количество членов в ряду.

}

BEGIN

writeln('Ввод: начальное значение X, конечное значение X, шаг X, точность вычислений.');

readln(xb, xe, xd, e);

x := xb;

writeln(' Таблица результатов вычислений ');

writeln('Знач. x | Знач. функции | Кол-во просуммированных членов');

while (x <= xe) do

begin

ch := power(x, 3) / 6;

done := true;

sum := x;

n := 1;

while (abs(ch) >= e) do

begin

{

Следующий if выполнится, если при n-ном номере члена

ряда заданная точность вычислений не будет достигнута.

}

if (n >= maxIter) then

begin

done := false;

break;

end;

{

1. Увеличение суммы на значение текущего члена.

2. Следующий член становится текущим.

3. Вычисление n-го члена ряда.

}

sum := sum + ch;

n := 1 + n;

ch := ch \* power(x \* (2 \* n - 1), 2) / (2 \* n \* (2 \* n + 1));

end;

{

Если заданная точность была достигнута, то условие

будет истинно. Если заданная точность достигнута не

будет, то условие будет ложно.

}

if done then

writeln(x,' | ' , sum, ' | ', n)

else

writeln(x,' | Ряд разошёлся | ', n);

x := x + xd;

end;

{ Конец работы программы. }

readln();

END.

* + 1. Тестирование работы программы

Результаты тестирования алгоритма продемонстрированы на Рисунках 3.2-3.4.

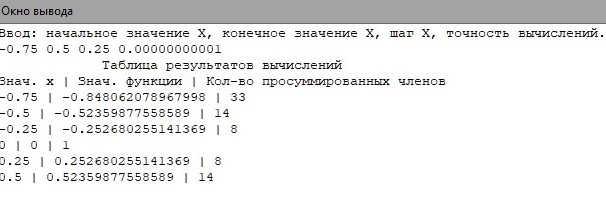


Рисунок 3.2 – Результат тестирования программы с точностью вычислений 0.00000000001

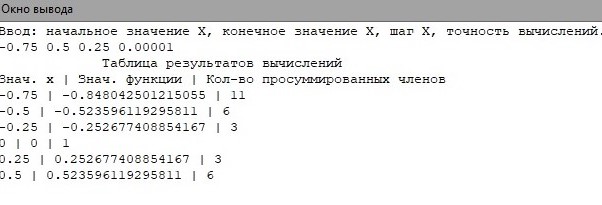


Рисунок 3.3 - Результат тестирования программы с точностью вычислений 0.00001

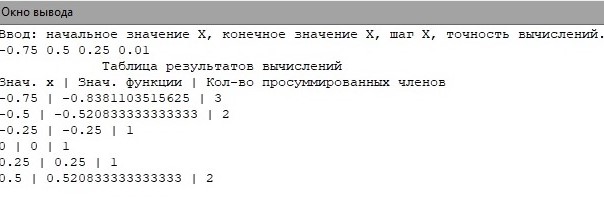


Рисунок 1.4 - Результат тестирования программы с точностью вычислений 0.01

В каждом тесте xb, xe, xd остаются постоянными. Значение точности e изменяется от меньшего к большему числу. Результаты тестирования показывают, что при изменении точности меняется количество просуммированных членов ряда, причём чем выше точность (меньше число), тем больше членов ряда необходимо просуммировать.

Также проведено тестирование области значений функции. Результат тестирования показан на Рисунке 3.5.

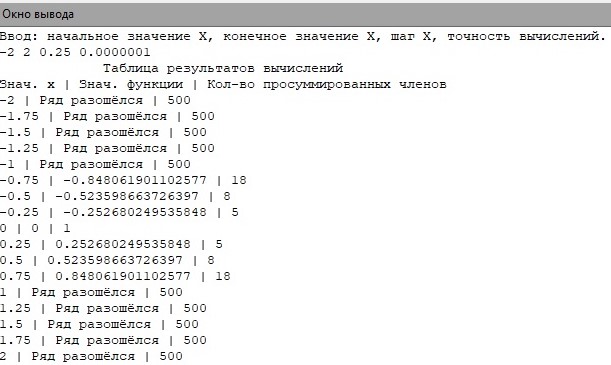


Рисунок 3.5 – Результат тестирования

Данный тест показывает, что область определения и область значений функции полностью соответствуют ожидаемым значениям.

Вывод

В ходе выполнения данной лабораторной работы были получены навыки программирования итерационных циклических структур. В ходе тестирования программы выявлено, что, во-первых, полученные значения функции полностью соответствуют ожидаемым, во-вторых, количество просуммированных членов ряда обратно пропорционально точности, т.е. при уменьшении числового значения точности увеличивается количество просуммированных членов ряда. Это связанно с тем, что при уменьшении точности можно просуммировать большее количество элементов, больших заданной точности. Полученные во время разработки навыки помогут разрабатывать более сложные циклические структуры, более эффективные по времени выполнения алгоритмы.