Филиал “Котельники” государственного бюджетного

образовательного учреждения высшего образования

Московской области «Университет «Дубна»

**Курсовая работа**

**по дисциплине: “Программирование на языке высокого уровня"**

**Вариант №19**

Выполнил:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

студент группы ИВТ-11 Третьяков Д.А.

Проверил:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

доцент, к.т.н. Артамонов Ю.Н.

Котельники – 2019

**Введение**

В связи с развитием информационных и телекоммуникационных технологий, начиная с XX века, мы смогли застать различные изменения в сферах жизни людей, а также в отрасли хозяйствования. В результате глобальных изменений в экономике и взятие курса на инновационное развитие, предприятия и организации во всем мире стали использовать различные средства ЭВМ(электронно-вычислительных машин), а позже и программного обеспечения. Но не смотря на эффективность электронных средств, они требовали определенного опыта и знаний, требуемых от человека. Возникла необходимость учета аппаратной организации каждой конкретной машины, то есть необходимость перекодировки программ при переносе с одной машины на другую – зачастую наблюдалась непереносимость алгоритмов, разработанных для одних машин при переносе на другие. Практически не представлялось возможным понять принципы построения чужой программы. Написанные на машинных кодах программы содержали минимум избыточной информации, которая бы позволяла обнаружить формальные ошибки кодирования. В результате, технические ошибки при набивке программы могли приводить к обескураживающим последствиям, а обнаружить такие ошибки было очень сложно. Так, например, для управления процессами ЭВМ в 1949 году был создан язык Ассемблер, который был более понятным, чем машинный язык. Спустя несколько десятилетий, люди открыли для себя множество других языков программирования «высокого уровня», таких как: ФОРТРАН (FORmula TRANslator – переводчик формул), АЛГОЛ (ALGOritmic Language – алгоритмический язык), БЕЙСИК (BASIC – Beginner’s Allpurpose Symbolic Instrucions Code – дословно: «многоцелевой код символических инструкций для начинающих»). В 1970 году Никлаус Вирт создал язык PASCAL (Паскаль). Этот язык обладает весьма развитыми средствами, особенно те его версии, которые используются в настоящее время. В настоящее время используется еще несколько мощных языков программирования, одним из которых является язык C/С++, созданный Бьёрном Страуструпом. Он предоставил возможность людям создавать программное обеспечение для решения не только организационных и экономических задач, но и повседневных. Основным отличием языка C++ от C(Си) в том, что у второго присутствует поддержка ООП(Объектно-ориентированного программирования).

Целью курсовой работы является изучение языка высокого уровня Си(C) для решения определенных задач: разработку игровых программ, а также решение численных алгоритмов и приближенных методов нахождения корней уравнений. Охарактеризовать процесс решения задач (разработка численных алгоритмов, приближенные методы нахождения корней уравнений, разработка игровых программ), предоставив методы решения и пояснения к каждой задаче (1. Дано; 2.Найти; Решение), описание входных данных(тип входных данных, ограничения, обработка ошибочного ввода, тестовые наборы входных данных), описание выходных данных(тип выходных данных, верификация выходных данных с использованием Wolfram (http://www.wolframalpha.com/), блок-схема реализуемого алгоритма, листинг программы на языке C, расчетные таблицы соответствия входных и выходных данных, выводы по результатам тестирования программного приложения на расчетных примерах.

**Глава I . РАЗРАБОТКА ЧИСЛЕННЫХ АЛГОРИТМОВ**

* 1. **Суммирование рядов и вычисление элементарных функций**

Целью данного задания является вычисление какого-либо выражения и подсчитать, сколько членов ряда и цепной дроби понадобится для нахождения суммы.

Дано: Соотношение С.Рамануджанова, имеющего вид:

Формула 1.1.1 - Соотношение С.Рамануджанова;

Найти: Вычислить, сколько членов ряда и цепной дроби нужно взять, чтобы достичь заданной точности.

Решение: При выполнении работы следует составить блок схему:

Наш ряд имеет вид:

Формула 1.1.2 – Нахождение соотношения Рамануджанова в виде ряда;

/\* Подключаем библиотеки \*/

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

/\* прототип функции для нахождения факториала \*/

double fctr(double);

/\* объявляем функцию для нахождения факториала \*/

double fctr(double n)

{

if (n == 1.0)

/\* условие, при котором программа

заканчивает свою работу, когда n == 1

и возвращает выч.значение \*/

return 1;

else return n \* fctr(n - 2.0); /\* работа рекурсивной функции \*/

}

int main()

{

int count = 0; /\* переменная для подсчета кол-во членов ряда \*/

double eps, fct = 2.0, n = 3.0, t = 1.0; /\* объявление переменных для работы \*/

printf("Введите точность вычисления: "); scanf("%lf", &eps); /\* вводим точность вычисления \*/

/\* Суммируем,пока член ряда t не станет достаточно малым по модулю \*/

while (eps < fabs(t))

{

t = 1.0 / fctr(n); /\* вычисляем n член ряда \*/

count++; /\* увеличиваем счетчик \*/

fct +=t; /\* суммируем \*/

n+=2; /\* увеличиваем значение n для след.члена ряда \*/

}

/\* выводим приближенное значение при заданной точности и кол-во членов ряда \*/

printf("%lf\nЧленов ряда: %d\n", fct, count);

/\* говорим о том, что программа завершила свою работу \*/

return 0;

}

Листинг 1.1.3 –Листинг программного кода

нахождения соотношения с помощью ряда;

|  |  |  |
| --- | --- | --- |
| № | Входные данные | Выходные данные |
| 1 | 0.1 | 2.400000; 2 |
| 2 | 0.01 | 2.409524; 3 |
| 3 | 0.001 | 2.410678; 5 |
| 4 | 0.0001 | 2.410678; 5 |
| 5 | 0.00001 | 2.410686; 6 |
| 6 | 0.000001 | 2.410686; 7 |
| 7 | 0.0000001 | 2.410686; 8 |

Листинг 1.1.4 –Таблица входных и выходных данных программы;

Аналогично делаем блок-схему для нахождения соотношения в виде цепной дроби:

Формула 1.1.5 – Нахождение соотношения Рамануджанова в виде цепной дроби;