**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**«Российский университет дружбы народов имени Патриса Лумумбы»**

Инженерная академия

*Департамент механики и процессов управления*

**ОТЧЕТ**

|  |  |
| --- | --- |
| **По** | **Механике космического полёта** |

|  |  |
| --- | --- |
| **Направление:** | **01.03.02 Прикладная математика и информатика** |
|  | (код направления / название направления) |
| **Профиль:** | **Математические методы механики полета ракет-носителей и космических аппаратов** |
|  | (название профиля) |

|  |  |
| --- | --- |
| **Тема:** | **Уравнение Кеплера (вариант 3)** |
|  | (название лабораторной / курсовой) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Выполнено студентом:** | **Прохорова Маргарита Вадимовна** | | |
|  | (ФИО) | | |
| **Группа:** |  | **ИПМбд-01-22** |  |
| **№ студенческого:** | | 1132226164 |  |

**Москва, 2023**

**Практическая часть (ч.1 — ч.4)**

#define \_USE\_MATH\_DEFINES

#include <iostream>

#include <fstream>

#include <cmath>

using namespace std;

double FixedPointIterations(double e, double M, double epsilon);

double BisectionMethod(double e, double M, double epsilon);

double GoldenSectionMethod(double e, double M, double epsilon);

double NewtonMethod(double e, double M, double epsilon);

int main() {

//Вспомогательные данные:

double epsilon = 0.00001;

//Начальные условия:

//Время полного оборота МКС вокруг Земли - 1 час 32 минуты и 53 секунды. Переводим в секунды:

//5573 сек.

double T = 5573.0; //период обращения МКС

double Rp = 415.0 + 6371.0; //радиус-вектор до перицентра МКС: перигей + радиус Земли

double Ra = 422.0 + 6371.0; //радиус-вектор до апоцентра МКС: апогей + радиус Земли

double e, n, М, E; //эксцентриситет, среднее угловое движение, средняя аномалия и эксц. аномалия МКС

e = (Ra - Rp) / (Ra + Rp); //расчёт эксцентриситета МКС

n = (2 \* M\_PI) / T; //расчёт среднего углового движения МКC

ofstream out; //поток для записи

out.open("values.txt"); //открываем файл для записи

//записываем в файл:

out << "t - arbitrary time, [c]\t";

out << "M - mean anomaly, [rad]\t";

out << "E - eccentric anonaly, [rad]\t";

out << "Theta - true anomaly, [rad]\n";

double Theta; //истинная аномалия

for (int t = 0; t <= T; t++) { //расчёт средней аномалии в зависимости от t (0 <= t <= T) при тау = 0

М = n \* t;

E = FixedPointIterations(e, М, epsilon);

Theta = 2 \* atan(sqrt(((1 + e) / 1 - e)) \* tan(E / 2));

if (Theta < 0) { //в случае, если ист. аномалия меньше 0, прибавляем период, равный два Пи.

Theta += 2 \* M\_PI;

}

//записывем в файл:

out << t << "\t";

out << М << "\t";

out << E << "\t";

out << Theta << "\n";

}

out.close(); //закрываем файл

return 0;

}

//метод итераций

double FixedPointIterations(double e, double M, double epsilon) {

double Ek = M;

double Ek1;

for (int i = 0; i <= 100; i++) {

Ek1 = e \* sin(Ek) + M;

if (abs(Ek1 - Ek) < epsilon) {

return Ek1;

}

Ek = Ek1;

}

return 0;

}

//метод половинного деления

double BisectionMethod(double e, double M, double epsilon) {

double a = M - 3; //выбранная граница отрезка

double b = M + 3; //выбранная граница отрезка

double c; //для произвольного отрезка, поделенного пополам

if ((a - e \* sin(a) - M) \* (b - e \* sin(b) - M) < 0) { //проверка для нулевой итерации

for (int i = 0; i <= 100; i++) {

c = (b + a) / 2;

if ((abs(c - e \* sin(c) - M) < epsilon) || ((abs(b-a)) < 2\*epsilon)) { //точность вычисления

return c;

}

if ((a - e \* sin(a) - M) \* (c - e \* sin(c) - M) < 0) {

b = c;

}

else {

a = c;

}

}

}

return 0;

}

//метод золотого сечения

double GoldenSectionMethod(double e, double M, double epsilon) {

double a = M - 3; //выбранная граница отрезка

double b = M + 3; //выбранная граница отрезка

double c; //отношение золотого сечения

double k = (sqrt(5) + 1) / 2; //константа для золотого сечения

if ((a - e \* sin(a) - M) \* (b - e \* sin(b) - M) < 0) { //проверка для нулевой итерации

for (int i = 0; i <= 100; i++) {

c = a + (b - a) / k;

if ((abs(c - e \* sin(c) - M) < epsilon) || ((abs(b - a)) < 2 \* epsilon)) { //точноcть вычисления

return c;

}

if ((a - e \* sin(a) - M) \* (c - e \* sin(c) - M) < 0) {

b = c;

}

else {

a = c;

}

}

}

return 0;

}

//метод Ньютона

double NewtonMethod(double e, double M, double epsilon) {

double Ek = M;

double Ek1, fk1, fk, val\_Ek, val\_Ek1;

for (int i = 0; i < 100; i++) {

val\_Ek = Ek; //сохранить значение Ek (т.к. потом меняется)

if (i == 0) {

//производную для нулевой итерации посчитаем самостоятельно:

Ek1 = Ek - ((Ek - e \* sin(Ek) - M) / (1 - e \* cos(Ek)));

}

else { //если не нулевая итерация, производная подсчитывается автоматически:

Ek1 = Ek - ((Ek - e \* sin(Ek) - M) / ((fk1 - fk) / (Ek1 - val\_Ek)));

}

if (abs(Ek1 - Ek) < epsilon) { //точность вычисления

return Ek1;

}

fk = (Ek - e \* sin(Ek) - M); //для вычисления производной

fk1 = Ek1 - e \* sin(Ek1) - M; //для вычисления производной

val\_Ek = Ek;

Ek = Ek1;

}

return 0;

}

**Теоретическая часть (ч.5)**

**Интересные факты об МКС**

1. Космонавты на борту МКС встречают закаты и рассветы по 16 раз в сутки, причина тому – огромная скорость, на которой движется станция, что позволяет делать полноценный оборот вокруг Земли за 93 минуты.

2. На данный момент это самый дорогой международный проект в истории, на создание которого потратили150 млрд долларов, хотя сама по себе конструкция занимает площадь футбольного поля.

**Расчёты радиус-вектора до апоцентра и радиус-вектора до перицентра**

Возьмем недостающие данные о высоте апогея и высоте перигея из Wikipedia (<https://ru.wikipedia.org/wiki/Международная_космическая_станция>):

Апогей = 422 км;

Перигей = 415 км;

Также нам недостаёт данных о среднем радиусе Земли. Также возьмем их из Wikipedia (<https://ru.wikipedia.org/wiki/Земля>):

Rз = 6371 км;

Для того, чтобы найти радиус-вектор до апоцентра и радиус-вектора до перицентра, необходимо к высоте апогея и высоте перигея прибавить радиус Земли:

***r*а** = Апогей + Rз = (422 + 6371) км = **6793 км**;

***r*п** = Перигей + Rз = (415 + 6371) км = **6786 км**;

**Расчёт большой полуоси**

Формула для вычисления большой полуоси:

a = (*r*a - *r*п) / 2;

Подставляем данные, полученные в предыдущем пункте:

**a** = (6793 км + 6786 км) / 2 = **6789,5 км;**

**Расчёт эксцентриситета**

Формула для вычисления эксцентриситета:

e = (*r*a – rп) / 2a;

Подставляем данные, полученные в предыдущем пункте:

**e** = (6793 км – 6786 км) / (2 \* 6789,5 км) = 7 / 13579 = **0,000515502**;

**Ответ для теоретической части:**

***r*а** = 6793 км

***r*п** = 6786 км

**a** = 6789,5 км

**e** = 0,000515502

**Продолжение практической части (ч.5)**

Сидерический период обращения — 2 часа = 120 мин = 7200 секунд. При этом наши значения t – от 0 до 5573 cекунд. Так что будем рассматривать весь промежуток времени.

**Программа на языке Python для построения графиков**

import matplotlib.pyplot as plt

i = 0

with open('values.txt', 'r') as f:

    line = f.readlines()

k = len(line)

time=[0]\*(k-1)

mean\_anomaly=[0]\*(k-1)

eccentric\_anomaly=[0]\*(k-1)

true\_anomaly=[0]\*(k-1)

for j in range (0, 4):

    for x in range (0, k-1):

        string = line[x+1].split()

        if (j==0):

            time[x] = float(string[0])

        if (j==1):

            mean\_anomaly[x] = float(string[1])

        if (j==2):

            eccentric\_anomaly[x] = float(string[2])

        if (j==3):

            true\_anomaly[x] = float(string[3])

x = time

#y = mean\_anomaly

y1 = eccentric\_anomaly

#y2 = true\_anomaly

#plt.plot(x, y, color='green')

plt.plot(x, y1, color='green')

#plt.plot(x, y2, color='green')

plt.xlabel('Время t [c]') #Подпись для оси х

#plt.ylabel('Средняя аномалия M(t) [рад]') #Подпись для оси y

plt.ylabel('Эксцентрическая аномалия E(t) [рад]') #Подпись для оси y

#plt.ylabel('Истинная аномалия Theta(t) [рад]') #Подпись для оси y

#plt.title('Метод итераций. Зависимость средней аномалии от времени') #Название

plt.title('Метод итераций. Зависимость эксцентрической аномалии от времени') #Название

#plt.title('Метод итераций. Зависимость истинной аномалии от времени') #Название

#plt.title('Метод половинного деления. Зависимость средней аномалии от времени') #Название

#plt.title('Метод половинного деления. Зависимость эксцентрической аномалии от времени') #Название

#plt.title('Метод половинного деления. Зависимость истинной аномалии от времени') #Название

#plt.title('Метод золотого сечения. Зависимость средней аномалии от времени') #Название

#plt.title('Метод золотого сечения. Зависимость эксцентрической аномалии от времени') #Название

#plt.title('Метод золотого сечения. Зависимость истинной аномалии от времени') #Название

#plt.title('Метод Ньютона. Зависимость средней аномалии от времени') #Название

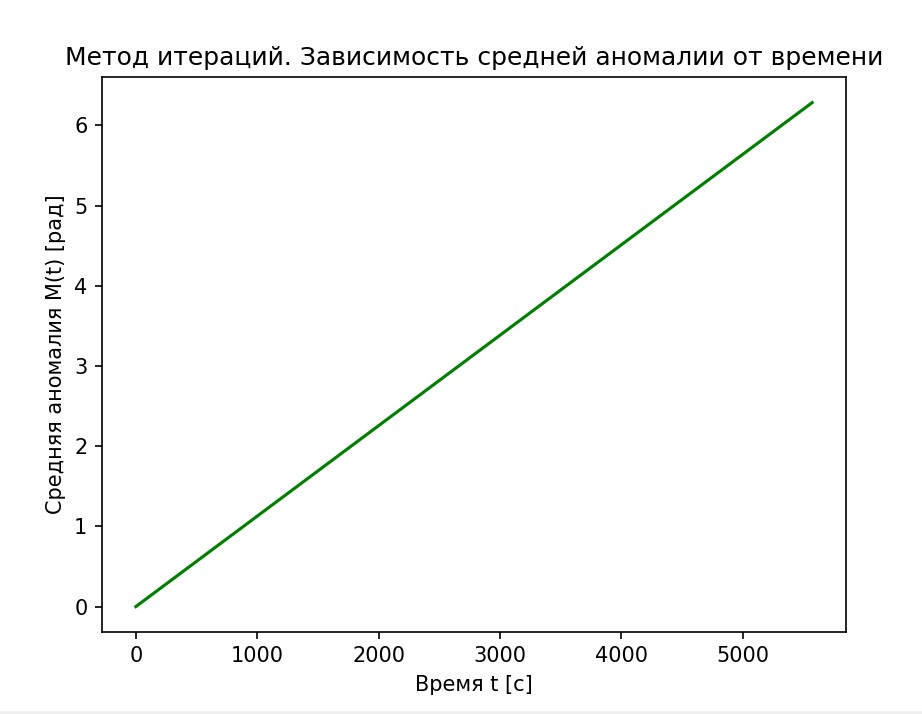
#plt.title('Метод Ньютона. Зависимость эксцентрической аномалии от времени') #Название

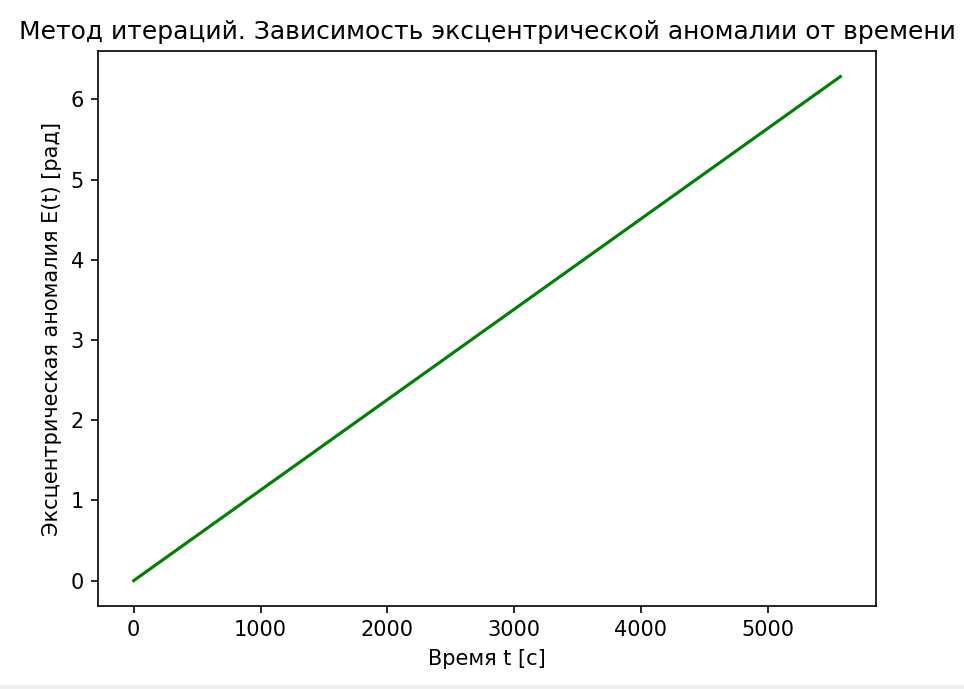
#plt.title('Метод Ньютона. Зависимость истинной аномалии от времени') #Название

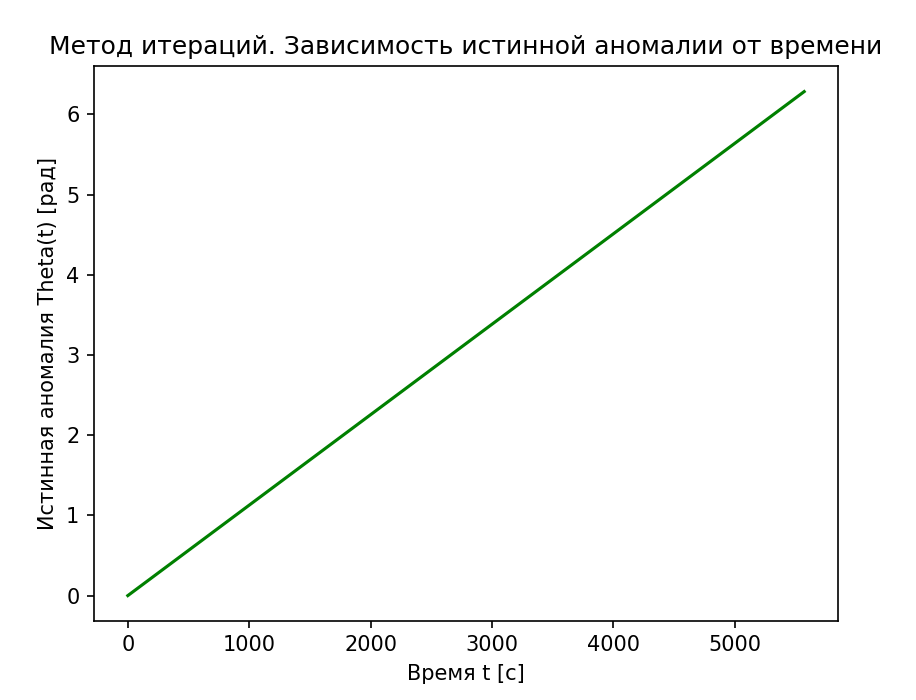
plt.show()

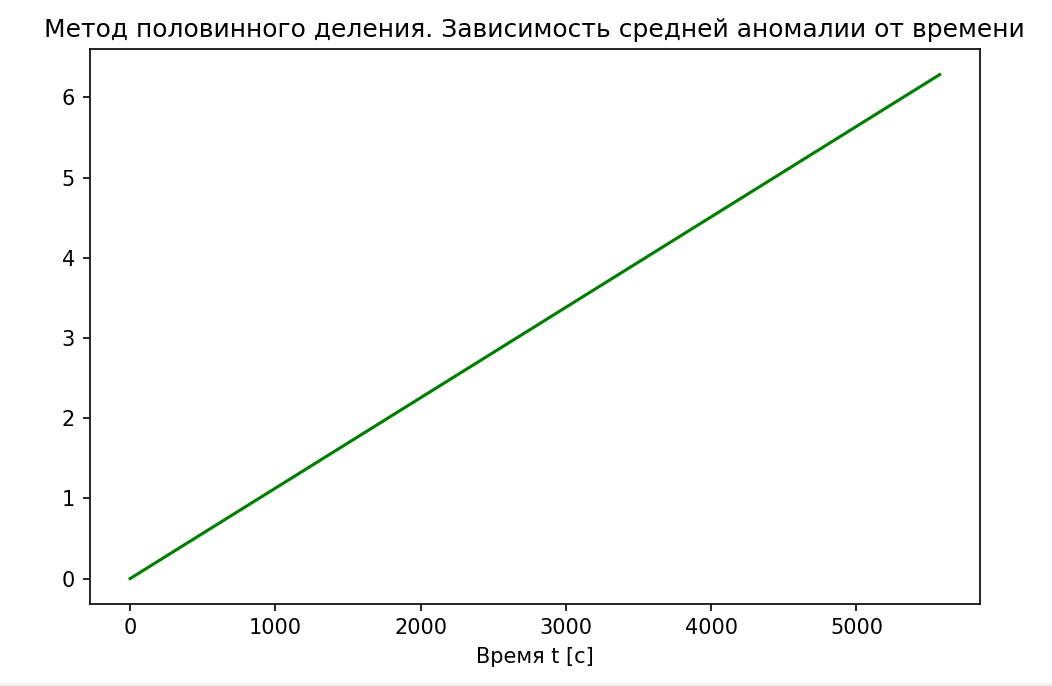
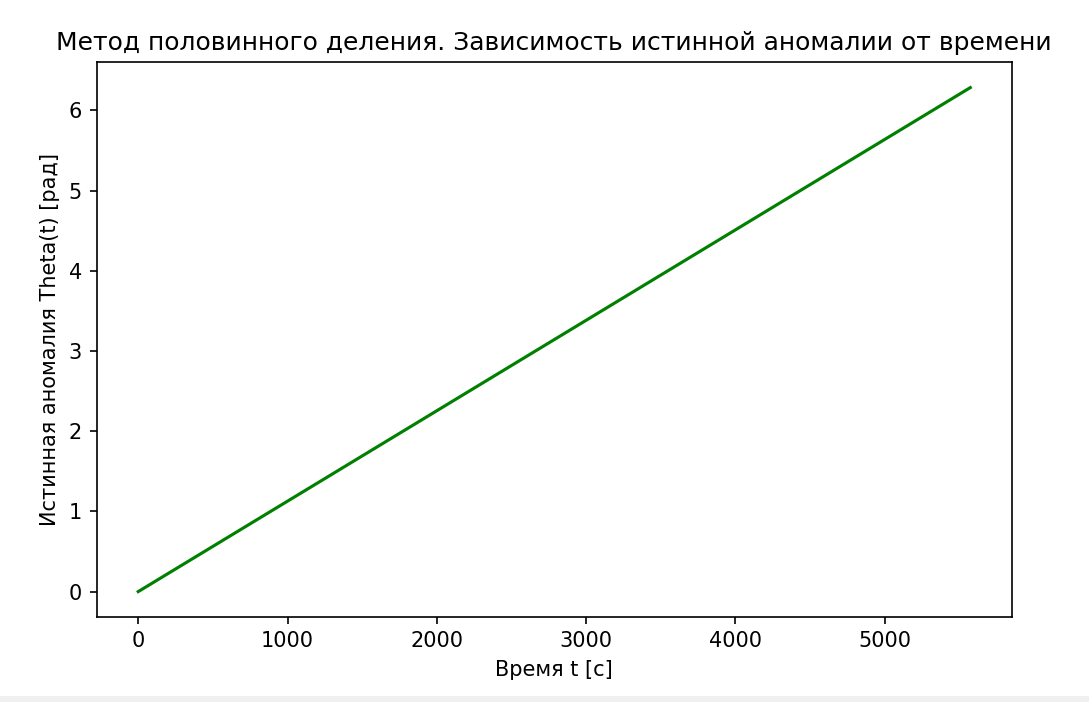
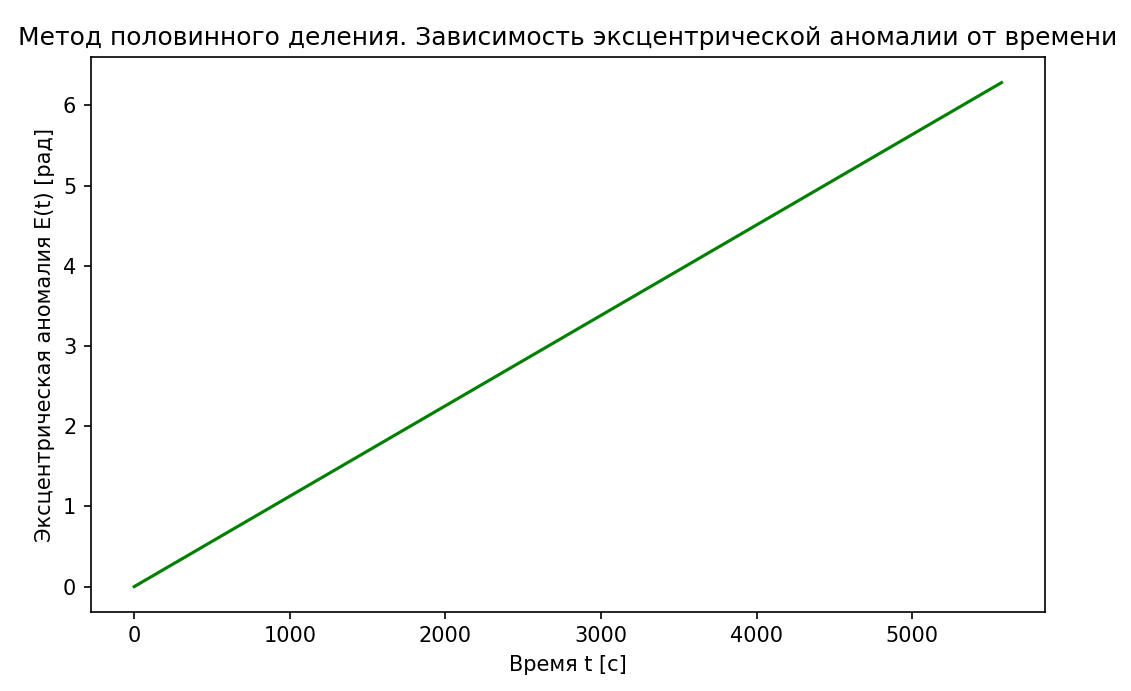
**Графики**

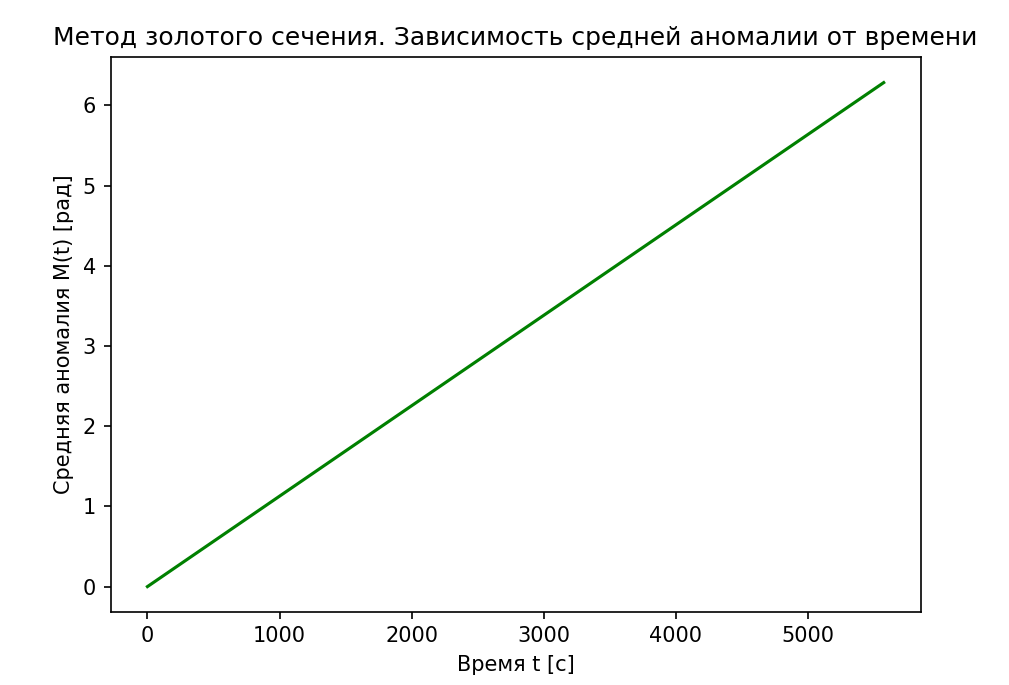
**1. Метод итераций**

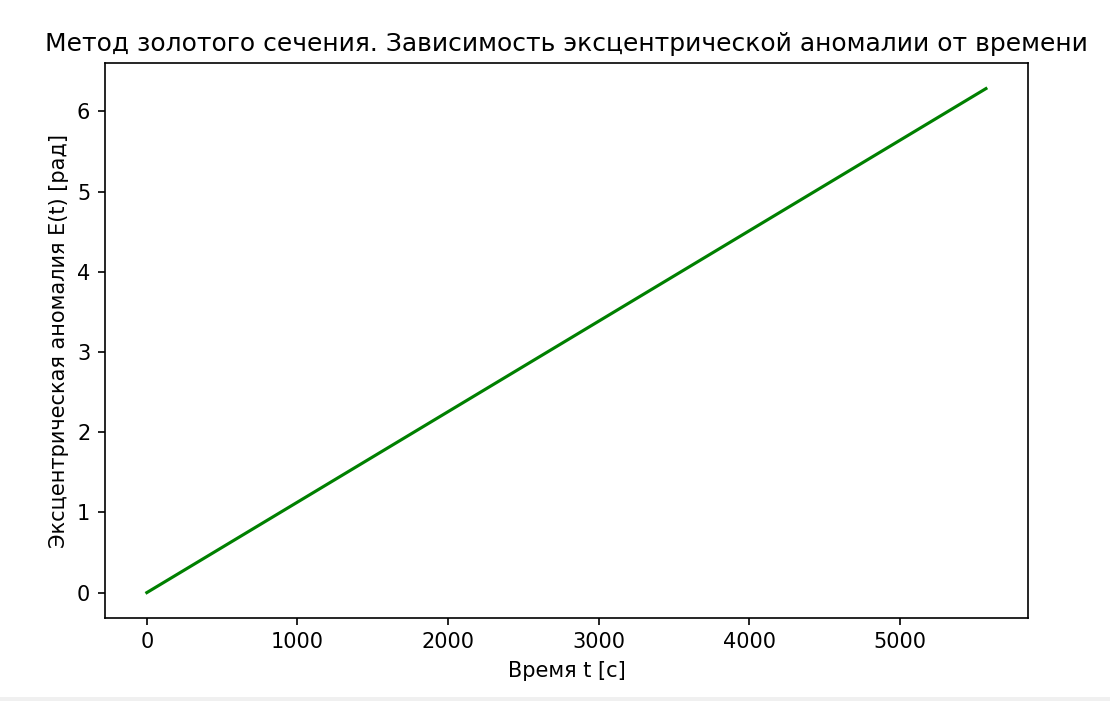
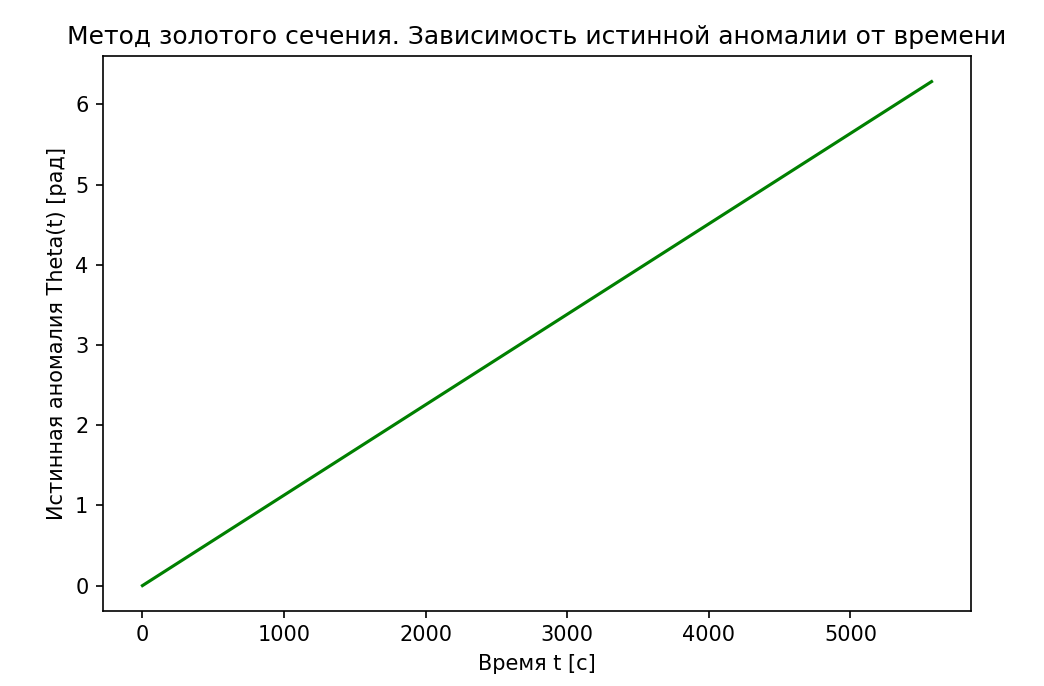
****





**2. Метод половинного деления  
  
  
  
** **3. Метод золотого сечения**



  
  
  
**4. Метод Ньютона**

