

Государственное бюджетное профессиональное образовательное учреждение города
Москвы "Московский колледж управления, гостиничного бизнеса и информационных
технологий "Царицыно" Отделение управления и информационных технологий.

Технические средства информатизации

Индивидуальный проект по теме 2:

Оценка производительности компьютера в тесте Молера на языках MATLAB, Scilab и Python

Студент: Уколов Алексей

Группа: ПЗ-3 Преподаватель: к.ф.-м.н. Мещеряков В.В.

Москва 2019

Аннотация:

Тест Молера производится для измерения производительности процессора
Intel(r) Core(tr) i3-3210 CPU @ 3.20 GHZ.

Оглавление

1 Оценка производительности компьютера в тесте Моллера.....	5
1.1 Тестирование в TestComp.....	5
1.2 Тестирование на языке Python.	8
1.3 Тестирование на языке Scilab.....	10
Заключение:	12
Литература и интернет источники:	13

Введение :

Ознакомиться с содержанием теста и построить графики числа гигафлопсов в зависимости от числа уравнений, используя Windows.

Технические средства :

1) Intel(r) Core(tr) i3-3210 CPU @ 3.20 GHZ.

2) Windows 10

1 Оценка производительности компьютера в тесте Моллера

1.1 Тестирование в TestComp.

Производительность компьютера определяется скоростью выполнения им определённых операций. Термин быстродействие используют для характеристики отдельных составляющих компьютера. Величина флопс (flops, акроним от англ. Floating-point Operations Per Second) – внесистемная единица, используемая для измерения производительности компьютеров – показывает, сколько операций с плавающей запятой в секунду выполняет данная вычислительная машина. Производные единицы образуют добавлением к слову флопс стандартных десятичных приставок системы СИ. Быстродействие составляющих ЭВМ определяется спецификационными параметрами вычислительного устройства. Например, 4-ядерный процессор Core 2 Quad, работающий на частоте 3.5 ГГц и выполняющий до 4-х операций за 1 такт в каждом ядре имеет предел скорости равный $4 \cdot 4 \cdot 3.5 = 56$ Gflops, а 2-х ядерный процессор Core 2 Duo, работающий на частоте 2.67 ГГц, имеет предел скорости равный $2 \cdot 4 \cdot 2.67 = 21.36$ Gflops. Производительность вычислительной системы оценивается специализированными тестовыми программами. Сегодня наиболее востребованы профессиональные программные библиотеки LINPACK (LINEar algebra PACKage) и LAPACK (Linear Algebra PACKage), содержащие наборы подпрограмм для решения систем линейных уравнений. Для оценки производительности в единицах GFLOPS (и объёма V кеш-памяти в мегабайтах используют формулы:

$$GFLOPS = \frac{\frac{2n^3}{3}}{T \times 10^9},$$

В этих формулах величина T -время в секундах, потребовавшихся для решений систем уравнений с матрицей коэффициентов $n \times n$, n_c - критическая длина матрицы, при которой вычислительная система выдаёт предупреждение Out of memory.

Формула матриц Гаусса систем линейных уравнений:

$$\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n = b_1 \\ a_{m1}x_1 + \dots + a_{mn}x_n = b_n \end{cases}$$

Её можно записать в матричном виде:

$$A = \begin{pmatrix} a_{10} & \dots & a_{1n} \\ a_{m1} & \dots & a_{mn} \end{pmatrix}, X = \begin{pmatrix} x_1 \\ x_n \end{pmatrix}, b = \begin{pmatrix} B_1 \\ B_m \end{pmatrix}$$

Матрица называется основной матрицей системы, b — столбцом свободных членов.

Простейший случай

В простейшем случае алгоритм выглядит так:

$$\begin{cases} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1n} \cdot x_n = b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2n} \cdot x_n = b_2 \\ a_{m1} \cdot x_1 + a_{m2} \cdot x_2 + \dots + a_{mn} \cdot x_n = b_m \end{cases}$$

Рис.1

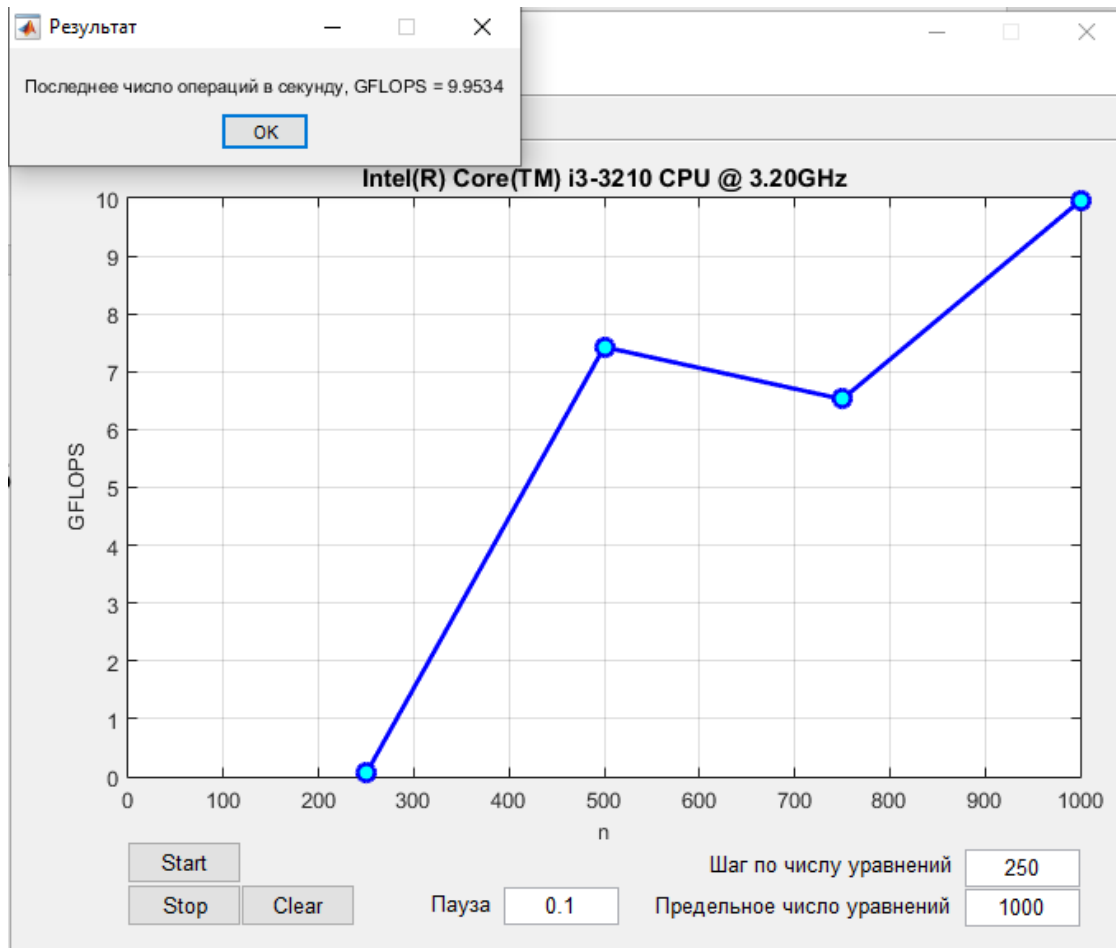


Рис.2

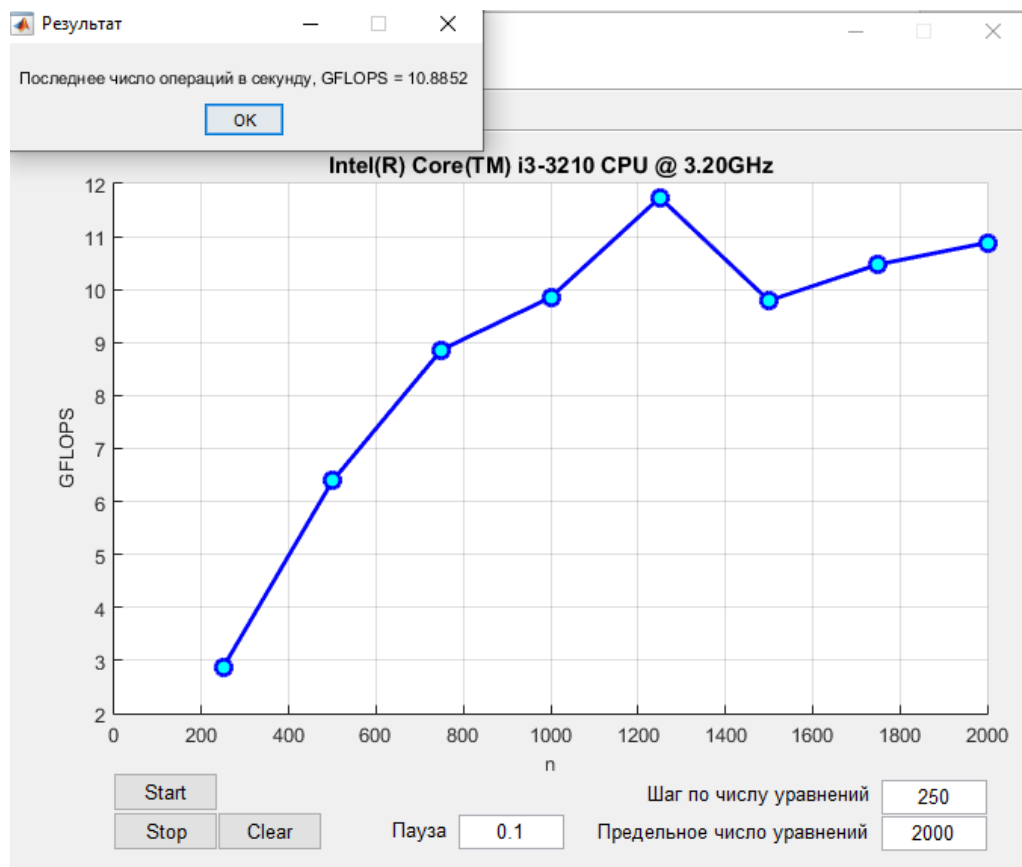


Рис.3

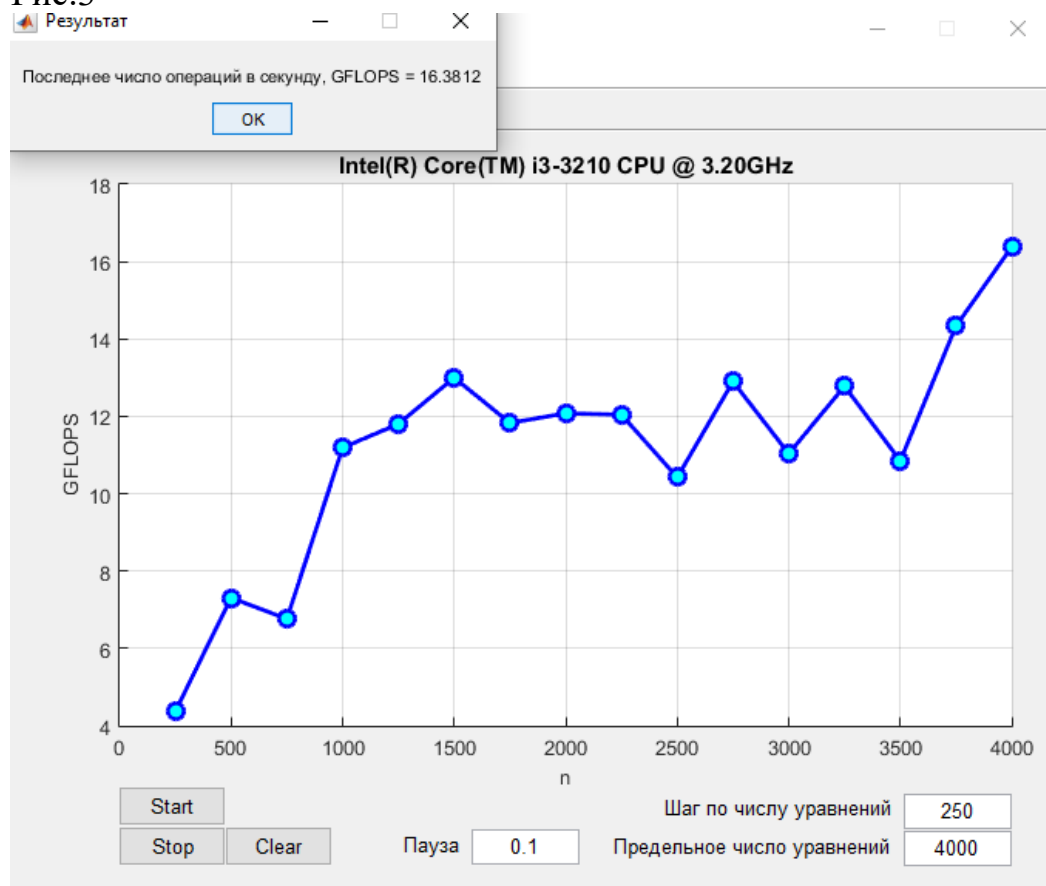


Рис.4



1.2 Тестирование на языке Python.

Следующий программный код иллюстрирует использование теста.

```
#from numpy.random import normal
import matplotlib.pyplot as plt
import time
#import guiwt.pyplot as plt # only this line has changed!
import numpy as np
N = 2000
dn = 250
GFLOPS = np.zeros(N//dn)
x = np.zeros(N//dn, dtype='int');
for k in range (N//dn):
    n=(k+1)*dn
    A = np.random.rand(n,n)
    b = np.random.rand(n)
    tic = time.time()
    np.linalg.solve(A,b)
    toc = time.time()
    T=toc-tic
    GFLOPS[k]=(2/3*n**3+2*n**2)/(T+1e-2)/1e9
    x[k] = n
    #print(x)
    #print(GFLOPS)
    plt.plot(x[0:k+1],GFLOPS[0:k+1], 'ro-')
    plt.xlim(0,N)
    plt.ylim(0,20)
```



```
plt.grid(True)
plt.show()
```

Результат исполнения данного листинга: на версиях 3.5.2.3

Рис.5

```
[ 250  500  750 1000 1250 1500 1750 2000]
[  0.81123832  4.65811275  7.43416332  9.98522781 10.88286474
 12.39473892 12.97535312 14.97068111]
```

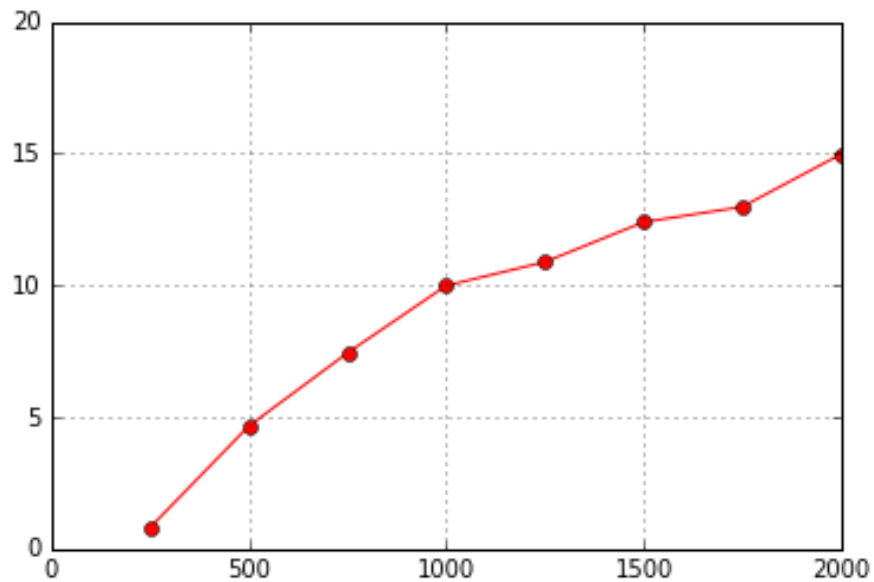


Рис.6

```
[ 250  500  750 ..., 3500 3750 4000]
[  0.81104487  3.49420993  7.43420998 ..., 16.00075065 16.31416591
 17.55337505]
```

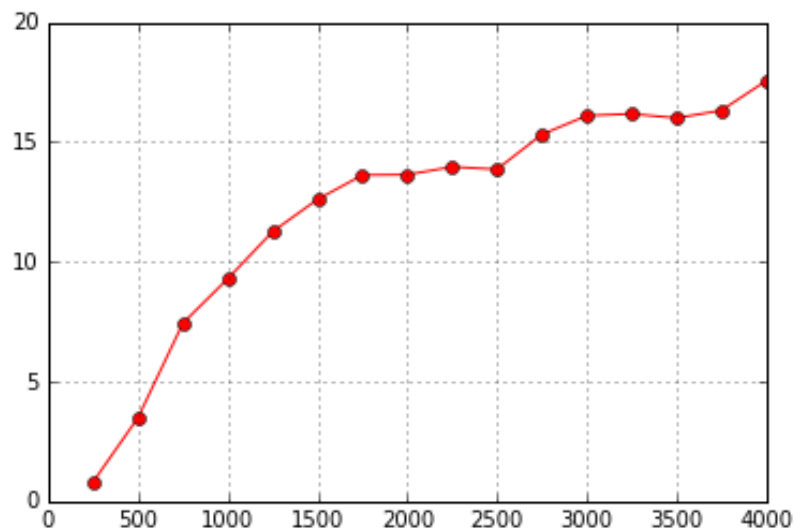
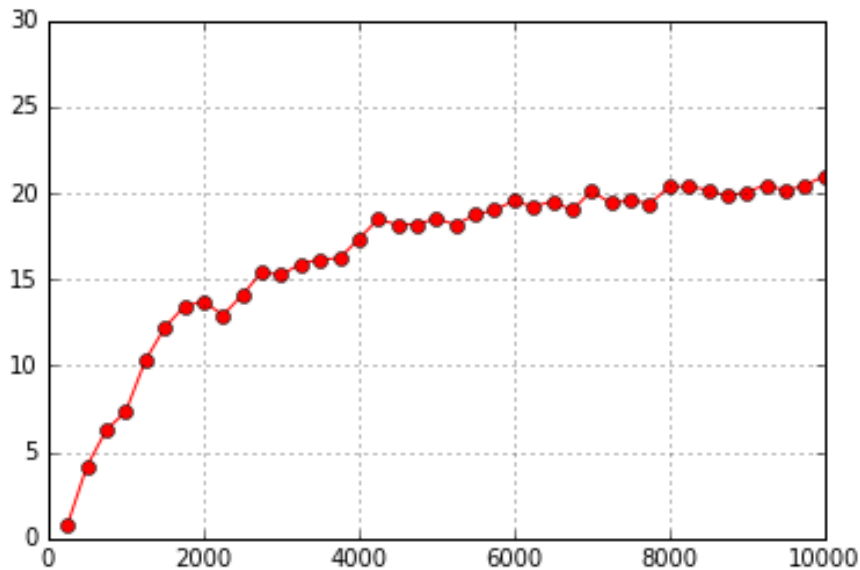


Рис.7

```
[ 250  500  750 ..., 9500 9750 10000]
[ 0.81095562  4.19216841  6.27788177 ..., 20.12933813  20.44079629
 20.98356834]
```



1.3 Тестирование на языке Scilab.

Следующий программный код иллюстрирует использование теста.

```
clear;clc
stacksize max
N = 2000;
dn = 250;
n = dn:dn:N
L = length(n)
GFLOPS = zeros (1,L,1)
Low = 0, High = 1

for k = 1:L
A=grand(n(k),n(k),"unf",Low,High)
b=grand(n(k),1,"unf",Low,High)
tic()
x = A\b
t = toc()
GFLOPS(k) = (2/3 * n(k)^3+2*n(k)^2)/1e9/t
end

plot(n,GFLOPS,'b.-','Marker','O',....
'Markersize',8,'MarkerFaceColor','c','LineWidth',2)
xgrid
xlabel('n'),ylabel('Gflops')
```

Результат исполнения данного листинга:

Рис.7

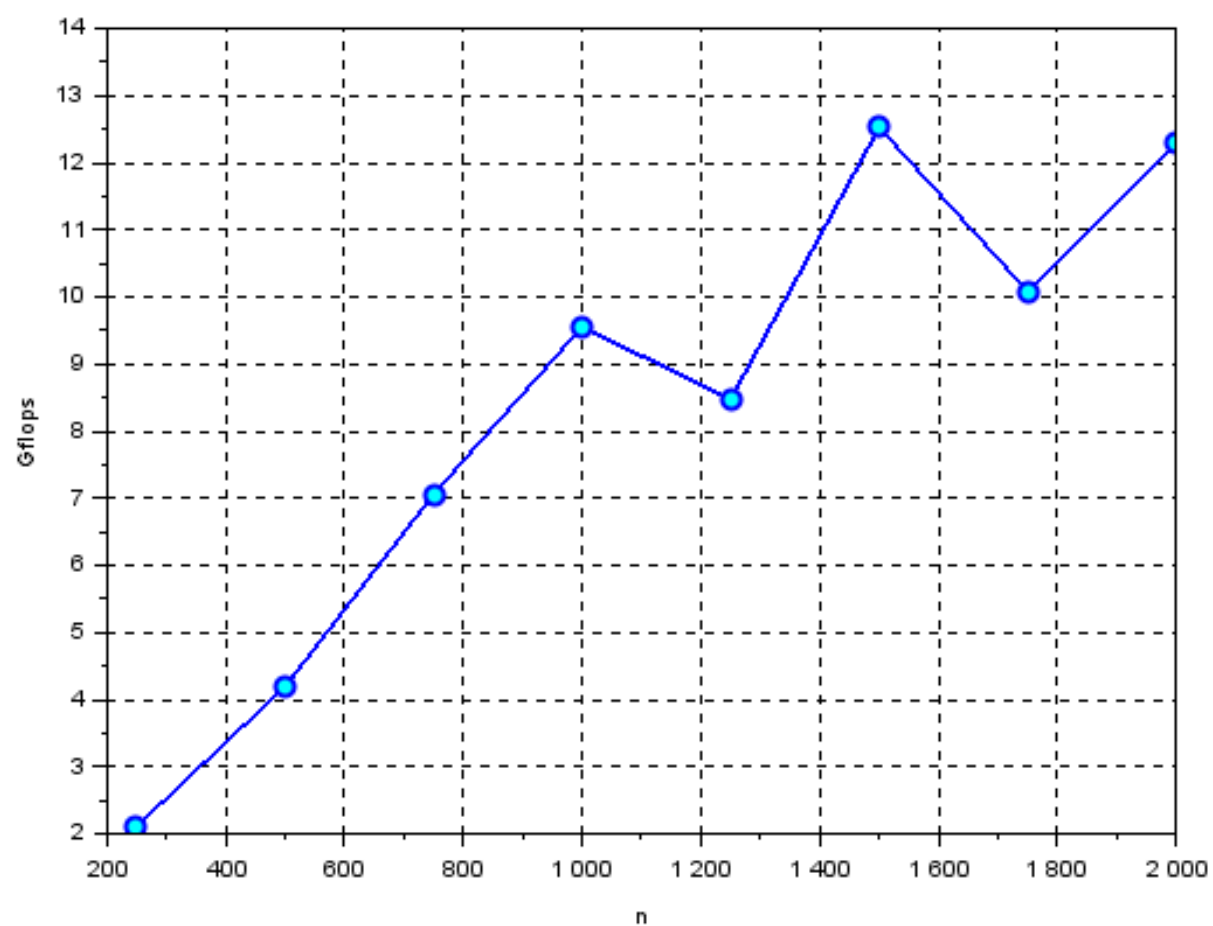
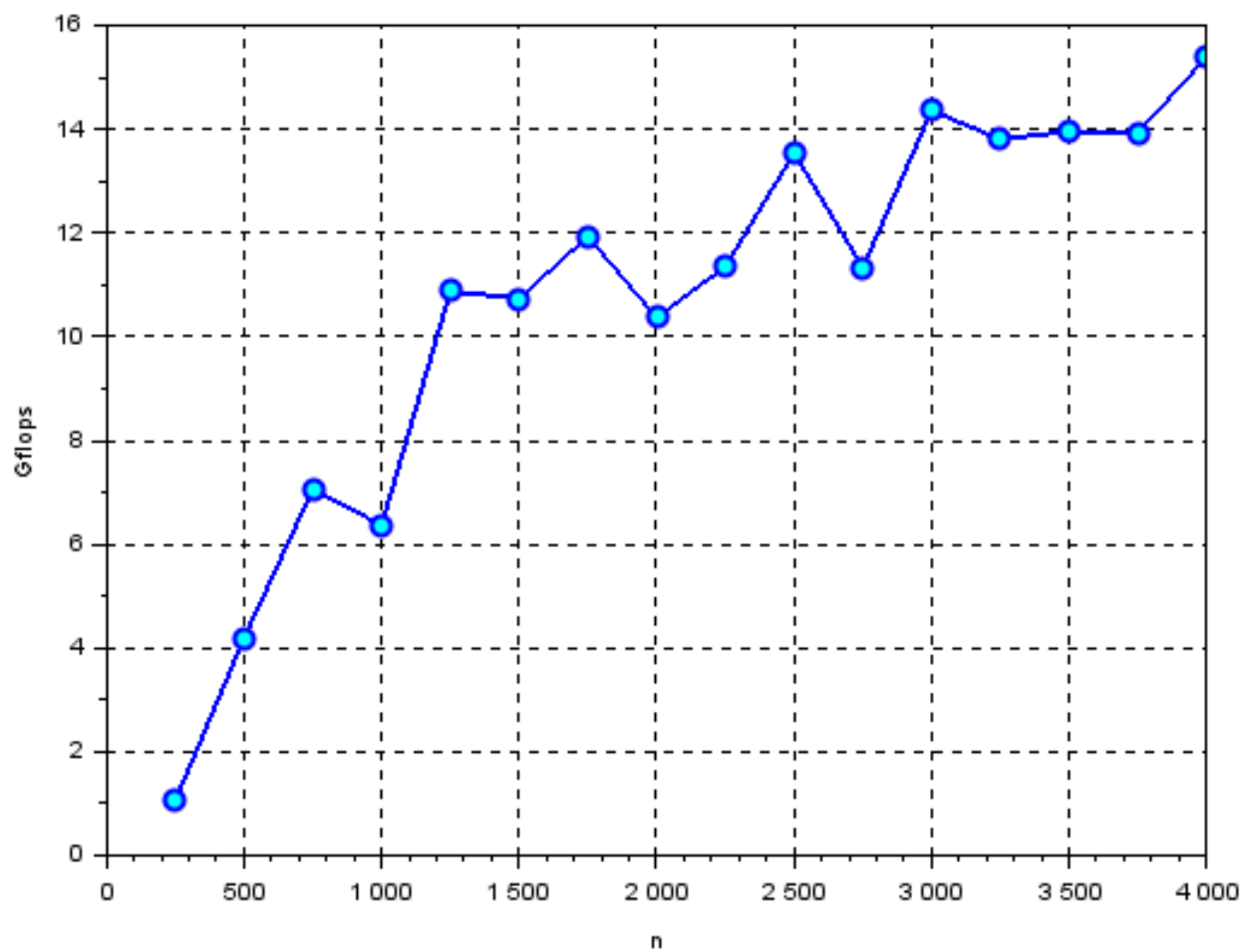


Рис.8



Заключение:

Литература и интернет источники:

1) <https://sourceforge.net>

2)

<https://sites.google.com/view/meshcher/%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D0%B0%D1%8F>

3) <https://www.mathworks.com/products/matlab.html>

4) <https://ru.wikipedia.org/wiki/FLOPS>

5)

https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%93%D0%B0%D1%83%D1%81%D1%81%D0%B0