

Московский Авиационный Институт
(национальный исследовательский университет)

ФАКУЛЬТЕТ № 8

Кафедра Вычислительной математики и
программирования

Отчет о лабораторных работах по дисциплине «Численные методы» за шестой семестр

Студент: Днепров И. С.

Группа: 8О-307Б-17

Преподаватель: Ревизников Д. Л.

Москва 2020 год

Оглавление

Лабораторная работа №1	2
Лабораторная работа №2	13
Лабораторная работа №3	21
Лабораторная работа №4	30

Лабораторная работа №1

Алгоритм: LUP-разложение.

Задачи, решаемые алгоритмом: Решение СЛАУ, нахождение определителя и обратной матрицы.

Алгоритм: Метод прогонки.

Задачи, решаемые алгоритмом: Решение СЛАУ с трехдиагональной матрицей.

Алгоритм: Метод простых итераций.

Задачи, решаемые алгоритмом: Решение СЛАУ.

Алгоритм: Метод Зейделя.

Задачи, решаемые алгоритмом: Решение СЛАУ.

Алгоритм: Метод Якоби (вращений).

Задачи, решаемые алгоритмом: Нахождение собственных значений и собственных векторов симметричной матрицы.

Алгоритм: QR-алгоритм.

Задачи, решаемые алгоритмом: Нахождение собственных значений произвольной матрицы.

Демонстрация работы программы

```
MacBook-Vanya-Original:1 vanyadneprov$ python3 task1.py
```

Лабораторная работа 1.1

алгоритм LUP - разложения матриц

Матрица

-7.000 3.000 -4.000 7.000

8.000 -1.000 -7.000 6.000

9.000 9.000 3.000 -6.000

-7.000 -9.000 -8.000 -5.000

-126 29 27 34

L

1.000 0.000 0.000 0.000

-0.778 1.000 0.000 0.000

0.889 4.500 1.000 0.000

-0.778 -5.000 -1.895 1.000

U

9.000 9.000 3.000 -6.000

-7.000 -2.000 -5.667 -9.667

8.000 -9.000 15.833 54.833

-7.000 10.000 -30.000 57.895

L * U

9.000 9.000 3.000 -6.000

-14.000 -9.000 -8.000 -5.000

-15.500 -10.000 -7.000 6.000

5.842 30.053 -34.000 7.000

Ответ

8.0000000000000007 -9.000000000000007 2.0000000000000018
-4.999999999999999

Ответ, почлученный при помощи numpy

8.0 -9.0 1.9999999999999993 -5.0

MacBook-Vanya-Original:1 vanyadneprov\$ python3 task2.py

Лабораторная работа 1.2

Метод прогонки

Матрица

8.000 -4.000 0.000 0.000 0.000

-2.000 12.000 -7.000 0.000 0.000

0.000 2.000 -9.000 1.000 0.000

0.000 0.000 -8.000 17.000 -4.000

0.000 0.000 0.000 -7.000 13.000

32 15 -10 1333 -76

Ответ

9.577747198741893 11.155494397483784 14.244348338903087
95.88814625516022 45.785924906624736

Ответ, почлученный при помощи numpy

9.57774719874188 11.15549439748376 14.244348338903052
95.88814625516021 45.785924906624736

MacBook-Vanya-Original:1 vanyadneprov\$ python3 task3.py

Лабораторная работа 1.3

28.000 9.000 -3.000 -7.000

-5.000 21.000 -5.000 -3.000

-8.000 1.000 -16.000 5.000

0.000 -2.000 5.000 8.000

-159 63 -45 24

Метод итераций

Ответ на 0-ой итерации

-5.591517857142858 2.746173469387755 6.776785714285714 1.9921875

Ответ на 1-ой итерации

-5.337138985058309 3.5668048469387754 6.402453364158164

-0.5489477040816322

Ответ на 2-ой итерации

-6.276304194948069 3.1752251801549702 5.532448637937318

-0.10983214086415849

Ответ на 3-ой итерации

-6.1338752033439965 2.807201228207322 6.1147811272136705

0.3360258963279188

Ответ на 4-ой итерации

-5.841724514211767 3.0434574908729584 6.15989577103743

-0.11993789745671357

Ответ на 5-ой итерации

-6.0268212638193335 3.0586210757980083 5.88609775733022

-0.08907048418015417

Ответ на 6-ой итерации

-6.053313921409018 2.9537700483339986 5.989239922840744
0.08584417061811456

Ответ на 7-ой итерации

-5.964832195434178 2.997007739000618 6.050593892043544
-0.004832439691965451

Ответ на 8-ой итерации

-5.994825537597096 3.019729103046235 5.980718944000888
-0.032369247777060295

Ответ на 9-ой итерации

-6.01649963678046 2.992017108985132 5.988530447808607
0.016982935761003848

Ответ на 10-ой итерации

-5.994417217396905 2.9957668029725593 6.013058055127114
0.005172747365903785

Ответ на 11-ой итерации

-5.995947065350371 3.0051772586070835 5.998560517436083
-0.009219583711306711

Ответ на 12-ой итерации

-6.004123245040524 2.9993051671092683 5.995415991428345
0.002193991254219263

Ответ на 13-ой итерации

-5.99971930681853 2.998240271700084 6.002703817731534
0.002691297134601367

Ответ на 14-ой итерации

-5.998471854005856 3.0010950688556584 6.000590700745083
-0.002129818157188268

Ответ на 15-ой итерации

-6.0008211515916425 3.0002002275821225 5.9986388006322855
-9.542075176227627e-05

Ответ с точностью 0.01 полученный методом простых итераций

-6.0008211515916425 3.0002002275821225 5.9986388006322855
-9.542075176227627e-05

Метод Зейделя

Ответ на 0-ой итерации

-5.913869844812925 4.518973214285714 1.875 3.0

Ответ на 1-ой итерации

-5.988613414115647 4.7421875 1.875 3.0

Ответ на 2-ой итерации

-6.060360863095238 4.7421875 1.875 3.0

Ответ на 3-ой итерации

-6.060360863095238 4.7421875 1.875 3.0

Ответ с точностью 0.01 полученный методом Зейделя

-6.060360863095238 4.7421875 1.875 3.0

Ответ, почлученный через numpy.linalg

-6.0 3.0000000000000004 6.0 -4.102404214899978e-16

MacBook-Vanya-Original:1 vanyadneprov\$ python3 task4.py

Лабораторная работа 1.4

Метод вращений

Матрица

-7.000 -6.000 8.000

-6.000 3.000 -7.000

8.000 -7.000 4.000

Собственные значения на 0-ой итерации

-11.208243919473801 3.0 8.2082439194738

Собственные значения на 1-ой итерации

-11.208243919473801 -3.7539574023076074 14.962201321781407

Собственные значения на 2-ой итерации

-11.55307332294064 -3.4091279988407677 14.962201321781407

Собственные значения на 3-ой итерации

-11.607802106076015 -3.4091279988407677 15.016930104916781

Собственные значения на 4-ой итерации

-11.607802106076015 -3.4126094330731696 15.020411539149183

Собственные значения на 5-ой итерации

-11.607818229408347 -3.4125933097408367 15.020411539149183

Ответ

Собственные значения

-11.607818229408347 -3.4125933097408367 15.020411539149183

Собственные векторы

0.906 0.099 0.412

0.190 0.774 -0.603

-0.379 0.625 0.683

Проверка

-10.513 -0.336 6.194

-2.211 -2.643 -9.062

4.397 -2.133 10.253

=

-10.513 -0.336 6.194

-2.211 -2.643 -9.062

4.397 -2.133 10.253

Ответ, полученный через numpy.linalg

Собственные значения

15.020411540086565 -11.607818230345726 -3.4125933097408376

Собственные векторы

0.412 -0.906 0.099

-0.603 -0.190 0.774

0.683 0.379 0.625

Лабораторная работа №2

Алгоритм: Метод простых итераций.

Задачи, решаемые алгоритмом: Решение нелинейных уравнений и систем нелинейных уравнений.

Алгоритм: Метод Ньютона .

Задачи, решаемые алгоритмом: Решение нелинейных уравнений и систем нелинейных уравнений.

Демонстрация работы программы

```
MacBook-Vanya-Original:2 vanyadneprov$ python3 task1.py
```

Метод простых итераций с точностью 0.001:

Ответ 0.7960978276053851 на 1-ой итерации с точностью 0.07890217239461494.

Ответ 0.7793048076796235 на 2-ой итерации с точностью 0.016793019925761543.

Ответ 0.7754950121255978 на 3-ой итерации с точностью 0.003809795554025719.

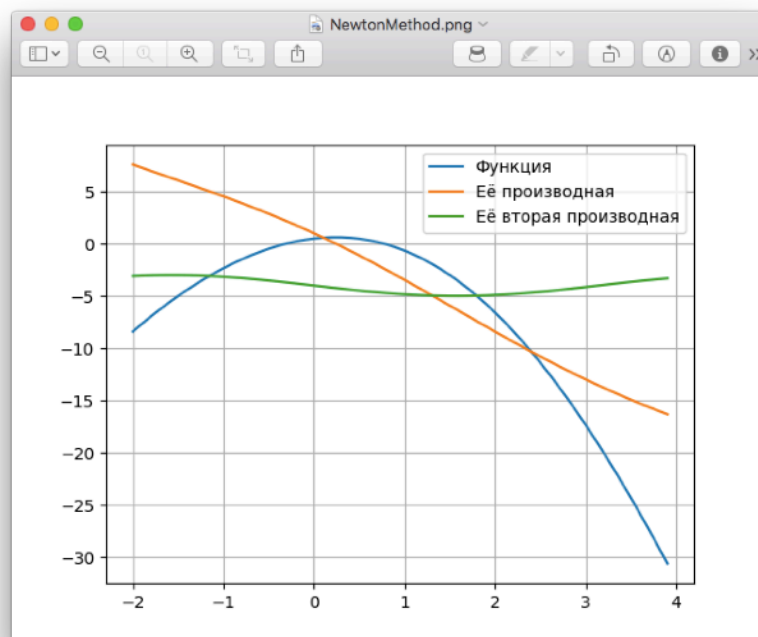
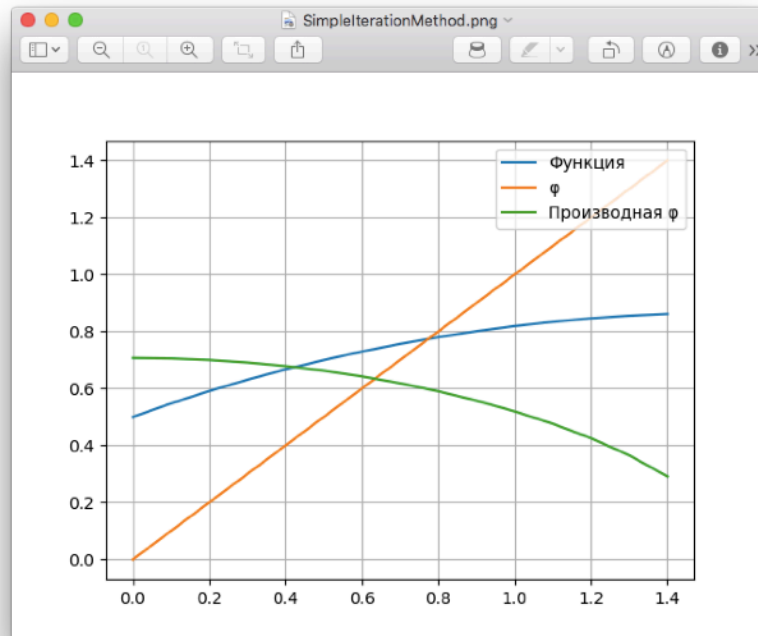
Ответ 0.7746191440568186 на 4-ой итерации с точностью 0.0008758680687791509.

Ответ 0.7744171786282515 на 5-ой итерации с точностью 0.00020196542856709687.

Метод Ньютона с точностью 0.001:

Ответ 0.7749695728448037 на 1-ой итерации с точностью 0.024969572844803722.

Ответ 0.774356963465415 на 2-ой итерации с точностью 0.0006126093793886733.



MacBook-Vanya-Original:2 vanyadneprov\$ python3 task2.py

Метод Ньютона с точностью 0.001:

На 1-ой итерации $x_1 = 0.809908384263519$, $x_2 = 1.751395732417488$ с точностью 0.30990838426351897.

На 2-ой итерации $x_1 = 0.8320743037840408$, $x_2 = 1.739508839978091$ с точностью 0.022165919520521826.

На 3-ой итерации $x_1 = 0.832187918786719$, $x_2 = 1.73940618182417$ с точностью 0.00011361500267825786.

Метод простых итераций с точностью 0.0001:

На 1-ой итерации $x_1 = 0.5838531634528576$, $x_2 = 1.479425538604203$ с точностью 0.8406317824339918.

На 2-ой итерации $x_1 = 1.0912437045717596$, $x_2 = 1.551242887914248$ с точностью 121.12795341342951.

На 3-ой итерации $x_1 = 1.0195521929040323$, $x_2 = 1.8872014237487185$ с точностью 1757.115005956214.

На 4-ой итерации $x_1 = 0.6888478739511618$, $x_2 = 1.851873569533319$ с точностью 0.6763550307064653.

На 5-ой итерации $x_1 = 0.7226092219926843$, $x_2 = 1.635648187326699$ с точностью 0.8265577334427391.

На 6-ой итерации $x_1 = 0.935193588511135$, $x_2 = 1.6613440532312254$ с точностью 100.91477237915784.

На 7-ой итерации $x_1 = 0.9095759546700333$, $x_2 = 1.804714019459514$ с точностью 34.853550171661766.

На 8-ой итерации $x_1 = 0.7682097100623418$, $x_2 = 1.7892434126968062$ с точностью 0.5525870206707695.

На 9-ой итерации $x_1 = 0.7832861243522968$, $x_2 = 1.6948488554615153$
с точностью 0.6193236930024761.

На 10-ой итерации $x_1 = 0.8762654012812382$, $x_2 = 1.7056117680677083$
с точностью 12.006367846836753.

На 11-ой итерации $x_1 = 0.8655925707722723$, $x_2 = 1.7683540057543994$
с точностью 6.851882353538435.

На 12-ой итерации $x_1 = 0.8037248965736783$, $x_2 = 1.7614794951576138$
с точностью 0.3465502062813044.

На 13-ой итерации $x_1 = 0.8104702749222147$, $x_2 = 1.719946268732854$
с точностью 0.3654130892240509.

На 14-ой итерации $x_1 = 0.8514024342433869$, $x_2 = 1.7246113480889822$
с точностью 3.6459030171868214.

На 15-ой итерации $x_1 = 0.8467907814483164$, $x_2 = 1.7522052489835096$
с точностью 2.3096391270043117.

На 16-ой итерации $x_1 = 0.8195844455162299$, $x_2 = 1.749158509873775$
с точностью 0.18262223056153476.

На 17-ой итерации $x_1 = 0.8225820218199404$, $x_2 = 1.7308622665242925$
с точностью 0.18595218287655862.

На 18-ой итерации $x_1 = 0.8406166962322448$, $x_2 = 1.7329049005992925$
с точностью 1.392773285488262.

На 19-ой итерации $x_1 = 0.838600507510461$, $x_2 = 1.7450546001554406$
с точностью 0.9145410936834493.

На 20-ой итерации $x_1 = 0.8266223081394425$, $x_2 = 1.7437082818428145$
с точностью 0.08755131400520419.

На 21-ой итерации $x_1 = 0.8279483937406897$, $x_2 = 1.735647643630155$
с точностью 0.08760095646102085.

На 22-ой итерации $x_1 = 0.835894334627348$, $x_2 = 1.7365452397390617$
с точностью 0.5781574550067883.

На 23-ой итерации $x_1 = 0.8350089736660439$, $x_2 = 1.741896472632976$
с точностью 0.3851091435167622.

На 24-ой итерации $x_1 = 0.8297334665547897$, $x_2 = 1.741302534622036$
с точностью 0.04008130179312347.

На 25-ой итерации $x_1 = 0.8303187619071727$, $x_2 = 1.737751467939321$
с точностью 0.039777115543548354.

На 26-ой итерации $x_1 = 0.8338193970595439$, $x_2 = 1.7381464583004138$
с точностью 0.24825910396580664.

На 27-ой итерации $x_1 = 0.8334299118812609$, $x_2 = 1.7405036009477464$
с точностью 0.16636712636969284.

На 28-ой итерации $x_1 = 0.8311061643131998$, $x_2 = 1.7402417903421097$
с точностью 0.017962826636957994.

На 29-ой итерации $x_1 = 0.8313642195982442$, $x_2 = 1.7386774429731784$
с точностью 0.017760593628670746.

На 30-ой итерации $x_1 = 0.8329063687740844$, $x_2 = 1.738851362882526$
с точностью 0.10814934211761325.

На 31-ой итерации $x_1 = 0.8327348965271923$, $x_2 = 1.7398896895406804$
с точностью 0.0726643580261809.

На 32-ой итерации $x_1 = 0.8317112882106436$, $x_2 = 1.739774324445786$
с точностью 0.007973476812620957.

На 33-ой итерации $x_1 = 0.8318250090592131$, $x_2 = 1.739085196959675$
с точностью 0.007870699673596592.

На 34-ой итерации $x_1 = 0.8325043612897829$, $x_2 = 1.7391617959040373$
с точностью 0.04740894842947302.

На 35-ой итерации $x_1 = 0.832428844964192$, $x_2 = 1.739619187985021$ с точностью 0.031889960292738224 .

На 36-ой итерации $x_1 = 0.831977937958859$, $x_2 = 1.7395683614558934$ с точностью 0.0035242856429890616 .

На 37-ой итерации $x_1 = 0.83202804211618$, $x_2 = 1.739264789141623$ с точностью 0.003476314745794566 .

На 38-ой итерации $x_1 = 0.8323273089415804$, $x_2 = 1.7392985290976823$ с точностью 0.02083945660396383 .

На 39-ой итерации $x_1 = 0.8322940467469688$, $x_2 = 1.7395000156303404$ с точностью 0.014024836830877888 .

На 40-ой итерации $x_1 = 0.8320954172610451$, $x_2 = 1.7394776245574368$ с точностью 0.001554806215996176 .

На 41-ой итерации $x_1 = 0.8321174904956113$, $x_2 = 1.7393438963476502$ с точностью 0.00153314777054649 .

На 42-ой итерации $x_1 = 0.8322493222031402$, $x_2 = 1.739358758694539$ с точностью 0.00917138757514117 .

На 43-ой итерации $x_1 = 0.8322346704823664$, $x_2 = 1.739447516098612$ с точностью 0.006173653579747743 .

На 44-ой итерации $x_1 = 0.8321471717015021$, $x_2 = 1.7394376522726342$ с точностью 0.0006853625170805356 .

На 45-ой итерации $x_1 = 0.8321568955883262$, $x_2 = 1.7393787430716265$ с точностью 0.0006757191917255967 .

На 46-ой итерации $x_1 = 0.8322149693781653$, $x_2 = 1.7393852900299047$ с точностью 0.004038439939181062 .

На 47-ой итерации $x_1 = 0.8322085152357386$, $x_2 = 1.7394243888524137$ с точностью 0.0027187107402952713 .

На 48-ой итерации $x_1 = 0.8321699708647285$, $x_2 = 1.7394200436526794$
с точностью 0.00030199881408028034 .

На 49-ой итерации $x_1 = 0.8321742544335038$, $x_2 = 1.7393940933221461$
с точностью 0.00029773088449841853 .

На 50-ой итерации $x_1 = 0.8321998367586028$, $x_2 = 1.7393969773260607$
с точностью 0.0017786620560420435 .

На 51-ой итерации $x_1 = 0.8321969936475881$, $x_2 = 1.7394142008888511$
с точностью 0.0011974610161016256 .

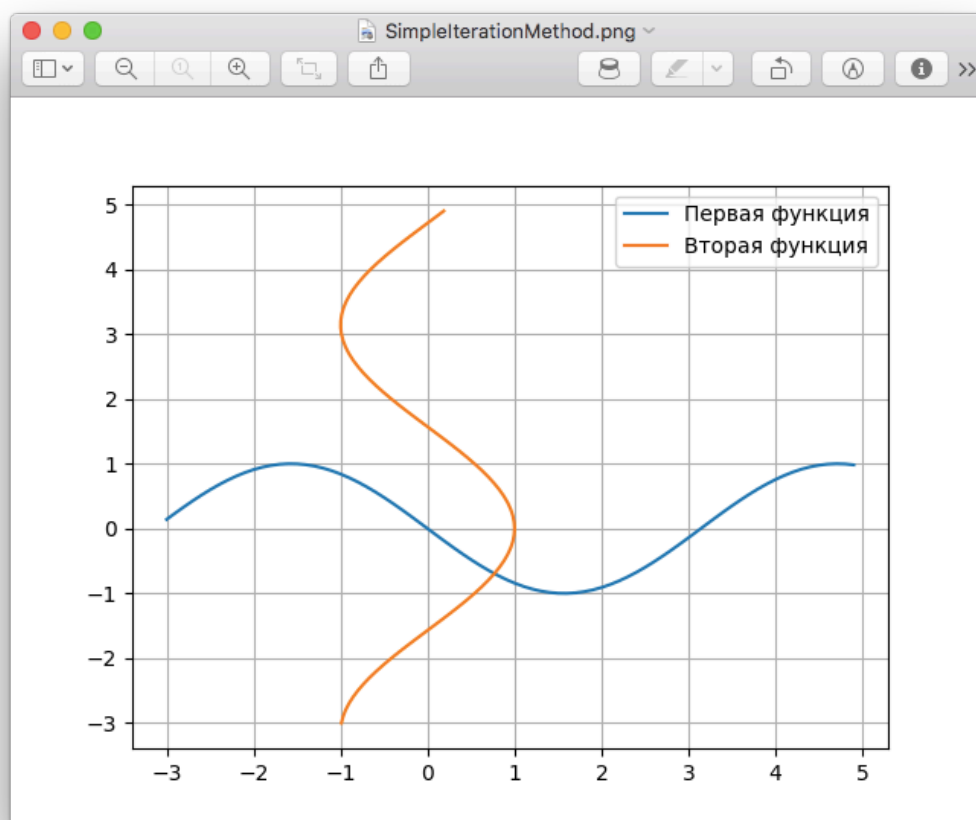
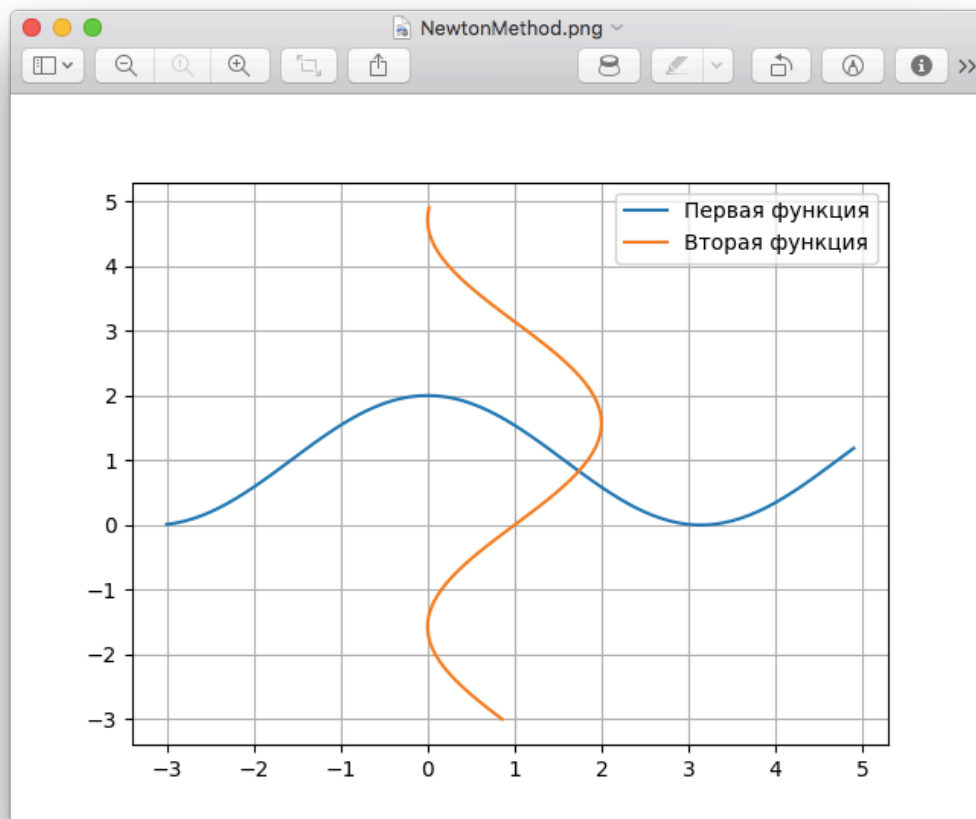
На 52-ой итерации $x_1 = 0.8321800143303856$, $x_2 = 1.739412286759098$
с точностью 0.0001330515612339707 .

На 53-ой итерации $x_1 = 0.8321819013136009$, $x_2 = 1.7394008552779308$
с точностью 0.00013116761335098324 .

На 54-ой итерации $x_1 = 0.8321931706842157$, $x_2 = 1.7394021257172307$
с точностью 0.0007834618169083972 .

На 55-ой итерации $x_1 = 0.8321919182600122$, $x_2 = 1.7394097129325095$
с точностью 0.0005274653765304923 .

На 56-ой итерации $x_1 = 0.8321844386386801$, $x_2 = 1.7394088697301595$
с точностью $5.861432998769449e-05$.



Лабораторная работа №3

Алгоритм: Построение интерполяционных многочленов Лагранжа и Ньютона.

Задачи, решаемые алгоритмом: Полиномиальная интерполяция.

Алгоритм: Построение кубического сплайна.

Задачи, решаемые алгоритмом: Сплайн-интерполяция.

Алгоритм: Метод наименьших квадратов (построение приближающих многочленов 1 и 2 степени).

Задачи, решаемые алгоритмом: Аппроксимация.

Алгоритм: Построение интерполяционного многочлена Ньютона и вычисление его 1 и 2 производной.

Задачи, решаемые алгоритмом: Численное дифференцирование.

Алгоритм: Метод прямоугольников.

Задачи, решаемые алгоритмом: Численное интегрирование.

Алгоритм: Метод трапеций.

Задачи, решаемые алгоритмом: Численное интегрирование.

Алгоритм: Метод Симпсона.

Задачи, решаемые алгоритмом: Численное интегрирование.

Демонстрация работы программы

MacBook-Vanya-Original:3 vanyadneprov\$ python3 task1.py

Полином Лагранжа, построенный по точкам $[-3, -1, 1, 3]$

```
LagrangePolynom(x) = 0.026 * (x + 1.0) * (x - 1.0) * (x - 3.0)
-0.049 * (x + 3.0) * (x - 1.0) * (x - 3.0) -0.049 * (x + 3.0) * (x
+ 1.0) * (x - 3.0) + 0.026 * (x + 3.0) * (x + 1.0) * (x - 1.0)
```

```
LagrangePolynom(-0.5) = -0.410
```

Оценка точности в заданной точке

```
|Arctg(x) - LagrangePolynom(x)| = 0.05364933
```

Полином Ньютона, построенный по точкам $[-3, -1, 1, 3]$

```
NewtonPolynom(x) = -1.249 + 0.232 * (x + 3.0) + 0.138 * (x + 3.0)
* (x + 1.0) -0.046 * (x + 3.0) * (x + 1.0) * (x - 1.0)
```

```
NewtonPolynom(-0.5) = -0.410
```

Оценка точности в заданной точке

```
|Arctg(x) - NewtonPolynom(x)| = 0.05364933
```

Полином Лагранжа, построенный по точкам $[-3, 0, 1, 3]$

$$\text{LagrangePolynom}(x) = 0.017 * x * (x - 1.0) * (x - 3.0) - 0.098 * (x + 3.0) * x * (x - 3.0) + 0.035 * (x + 3.0) * x * (x - 1.0)$$

$$\text{LagrangePolynom}(-0.5) = -0.410$$

Оценка точности в заданной точке

$$|\text{Arctg}(x) - \text{LagrangePolynom}(x)| = 0.05364933$$

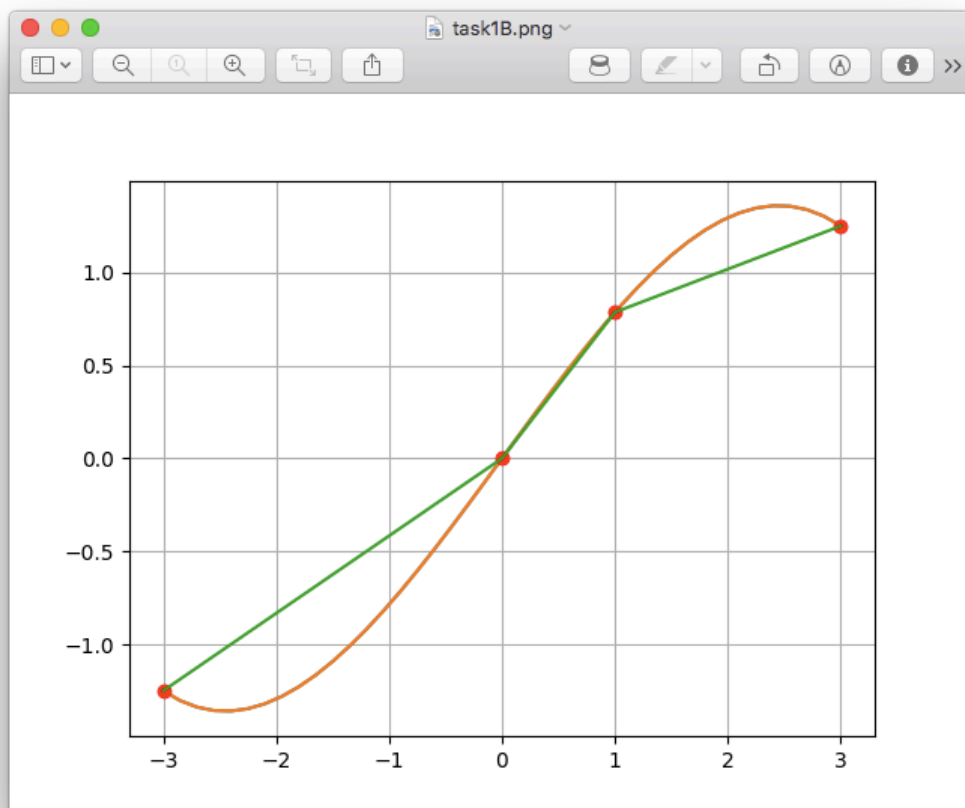
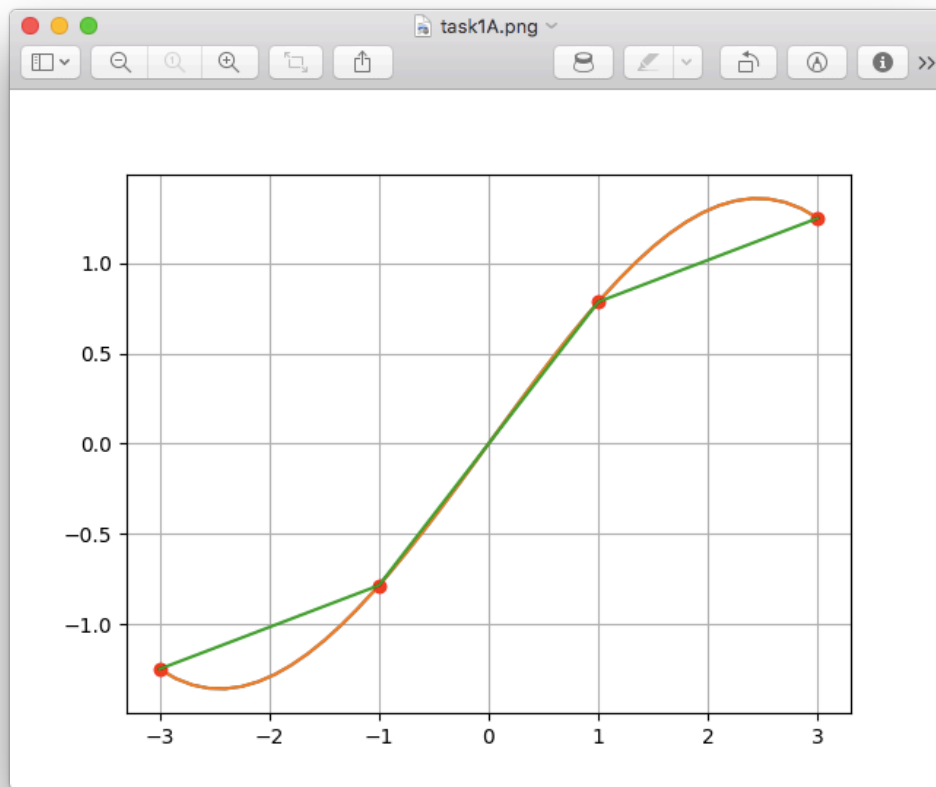
Полином Ньютона, построенный по точкам $[-3, 0, 1, 3]$

$$\text{NewtonPolynom}(x) = -1.249 + 0.416 * (x + 3.0) + 0.092 * (x + 3.0) * x - 0.046 * (x + 3.0) * x * (x - 1.0)$$

$$\text{NewtonPolynom}(-0.5) = -0.410$$

Оценка точности в заданной точке

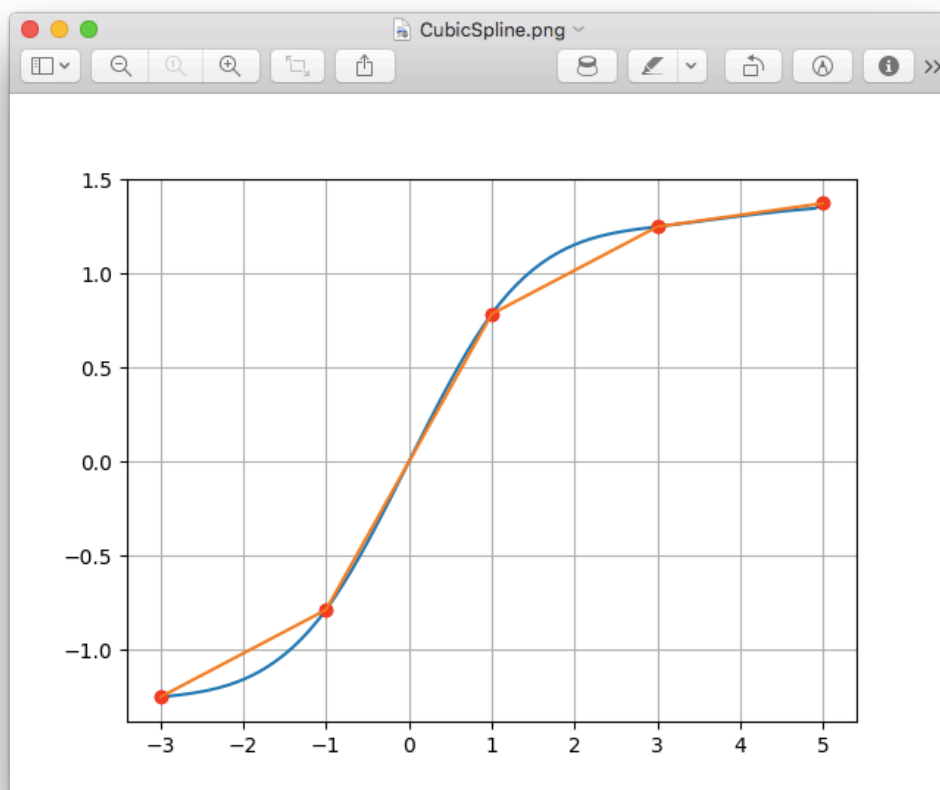
$$|\text{Arctg}(x) - \text{NewtonPolynom}(x)| = 0.05364933$$



MacBook-Vanya-Original:3 vanyadneprov\$ python3 task2.py

i.	$[x_{i-1}, x_i]$	a_i	b_i	c_i	d_i
1.	$[-3, -1]$	-1.2490	0.0470	0.0000	0.0462
2.	$[-1, 1]$	-0.7854	0.6014	0.2772	-0.0926
3.	$[1, 3]$	0.7854	0.5990	-0.2784	0.0474
4.	$[3, 5]$	1.2490	0.0542	0.0060	-0.0040

Spline(-0.5) = -0.427



```
MacBook-Vanya-Original:3 vanyadneprov$ python3 task3.py
```

Приближающего многочлена 1-ой степени:

0.0000		1.7000		3.4000		5.1000		6.8000		8.5000	
0.4713		1.0114		1.5515		2.0916		2.6317		3.1718	

Сумма квадратов ошибок: 0.487174

Приближающего многочлена 2-ой степени:

0.0000		1.7000		3.4000		5.1000		6.8000		8.5000	
0.1294		1.0797		1.8249		2.3651		2.7001		2.8300	

Сумма квадратов ошибок: 0.094558

Приближающего многочлена 3-ой степени:

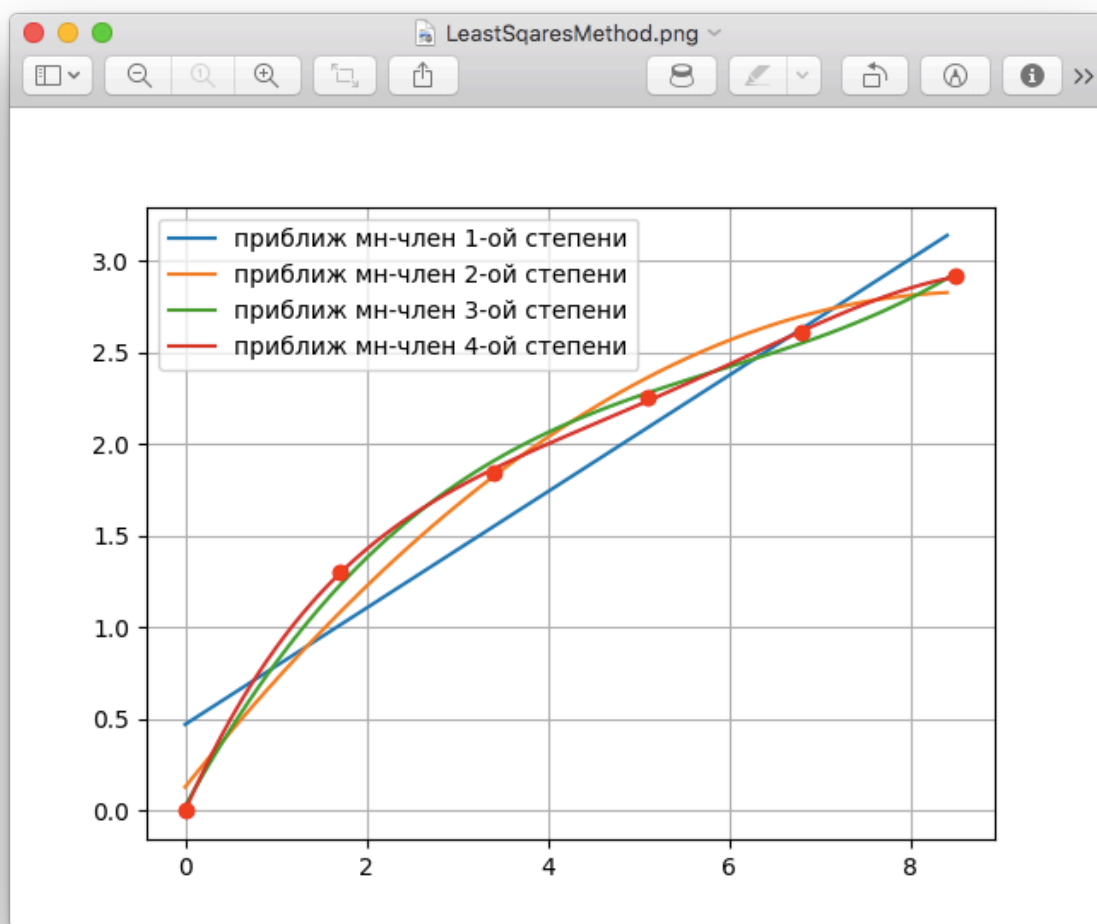
0.0000		1.7000		3.4000		5.1000		6.8000		8.5000	
0.0241		1.2272		1.9092		2.2808		2.5526		2.9353	

Сумма квадратов ошибок: 0.014648

Приближающего многочлена 4-ой степени:

0.0000		1.7000		3.4000		5.1000		6.8000		8.5000	
0.0021		1.2931		1.8653		2.2369		2.6184		2.9134	

Сумма квадратов ошибок: 0.001157



MacBook-Vanya-Original:3 vanyadneprov\$ python3 task4.py

Численное дифференцирование функции

$F'(1.0) = 1.51807.$

$F''(1.0) = -0.8242799999999999.$

MacBook-Vanya-Original:3 vanyadneprov\$ python3 task5.py

Численное дифференцирование

Метод прямоугольников

Ответ с шагом 0.5: $F(x) = 0.14373851198488408$

Ответ с шагом 0.25: $F(x) = 0.14499262372828325$

Погрешность согласно методу Рунге-Ромберга равна
0.0016721489911989036

Метод трапеций

Ответ с шагом 0.5: $F(x) = 0.1987479080146284$

Ответ с шагом 0.25: $F(x) = 0.17124320999975623$

Погрешность согласно методу Рунге-Ромберга равна
0.03667293068649624

Метод Симпсона

Ответ с шагом 0.5: $F(x) = 0.1453893675489163$

Ответ с шагом 0.25: $F(x) = 0.1454083106614655$

Погрешность согласно методу Рунге-Ромберга равна
 $2.5257483398951248e-05$

Лабораторная работа №4

Алгоритм: Метод Эйлера.

Задачи, решаемые алгоритмом: Задача Коши для ОДУ 2-го порядка.

Алгоритм: Метод Рунге-Кутты 4-го порядка.

Задачи, решаемые алгоритмом: Задача Коши для ОДУ 2-го порядка.

Алгоритм: Метод Адамса 4-го порядка.

Задачи, решаемые алгоритмом: Задача Коши для ОДУ 2-го порядка.

Алгоритм: Метод стрельбы.

Задачи, решаемые алгоритмом: Краевая задача для ОДУ 2-го порядка.

Алгоритм: Конечно-разностный метод.

Задачи, решаемые алгоритмом: Краевая задача для ОДУ 2-го порядка.

Демонстрация работы программы

```
MacBook-Vanya-Original:4 vanyadneprov$ python3 task1.py
```

В задании $y(0)$ и $y'(0)$ указаны не верно.

Я никак не мог понять, что за фигня творится с точностью.

И когда я уже отчаялся что-то починить, я пошёл строить график и всё понял.

Я пересчитал их значения:

$$y(0) = 1.8414709848078965$$

$$y'(0) = -0.0$$

Явный метод Эйлера

Абсолютная погрешность: 0.0516899253052536

Погрешность согласно методу Рунге-Ромберга: 0.0374385598140358

Явный итерационный метод Эйлера-Коши

Абсолютная погрешность: 0.06678896600015742

Погрешность согласно методу Рунге-Ромберга: 0.005299060918064344

Метод Рунге-Кутты

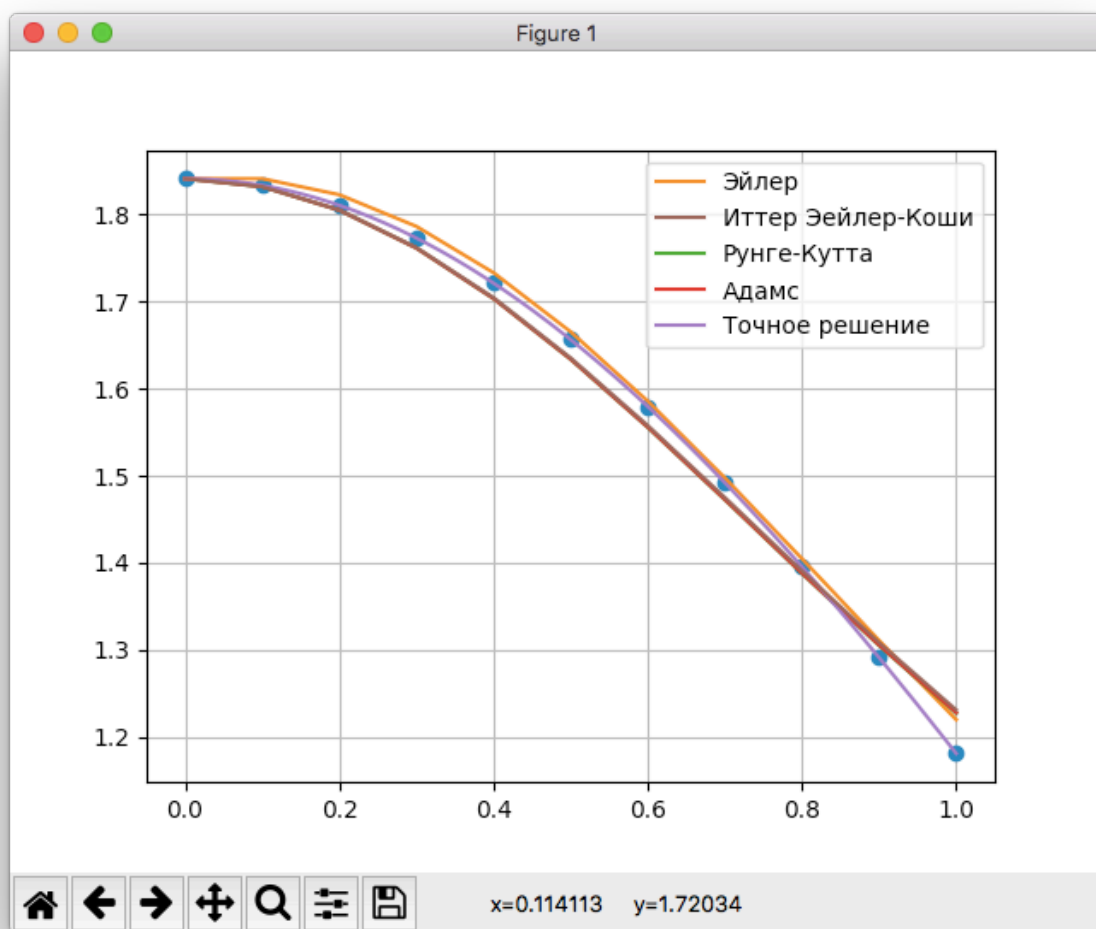
Абсолютная погрешность: 0.06593068847818892

Погрешность согласно методу Рунге-Ромберга: 2.3650732774339048e-07

Метод Адамса

Абсолютная погрешность: 0.06611016977822881

Погрешность согласно методу Рунге-Ромберга: 5.564820467043325e-05



MacBook-Vanya-Original:4 vanyadneprov\$ python3 task2.py

Метод стрельбы

Точность результата согласно методу Рунге-Ромберга =
0.0023826733588001503

Конечно-разностный метод

Точность результата согласно методу Рунге-Ромберга =
0.0630052875933154

