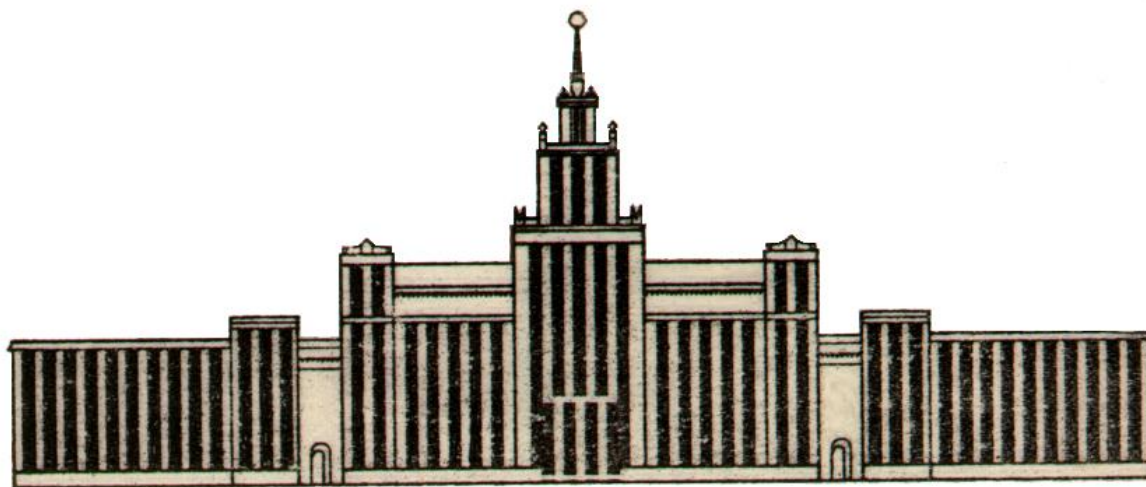

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ



ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УДК 681.3.(07)

К721

В. В. Костерин

PYTHON: КУРС МОЛОДОГО БОЙЦА

Методические указания к курсовой работе по программированию

Челябинск
2019

Министерство науки и высшего образования Российской Федерации

Южно-Уральский государственный университет
(национальный исследовательский университет)

Высшая школа экономики и управления
Кафедра «Информационные системы»

УДК 681.3.(07)

K721

В. В. Костерин

PYTHON: КУРС МОЛОДОГО БОЙЦА

Методические указания к курсовой работе по программированию

Челябинск
Издательство ЮУрГУ
2019

УДК 681.3.05(075.8) + 681.3.068(075.8)

К721

*Одобрено
учебно-методической комиссией
Высшей школы экономики и управления*

*Рецензенты:
И.Б. Фетисов, Д.В. Морозов*

Костерин, В.В.

К721 **Python: Курс молодого бойца:** Методические указания к курсовой работе по программированию / В.В. Костерин – Челябинск: Изд-во ЮУрГУ, 2019 – XX с.

Комплекс практических работ состоит из трёх частей: первая – разминка, для получения первоначальных навыков программирования. Вторая – экспериментальное исследование трех базовых алгоритмов сортировки, вычисление показателей эффективности этих алгоритмов на различных выборках и сравнение полученных результатов с теоретическими вычислениями. Третья часть посвящена работе с популярными пакетами Python – NumPy, матричная алгебра и линейный анализ, и Matplotlib для визуализации результатов вычислений. Здесь же надо вспомнить линейную алгебру и решить несколько систем линейных уравнений.

Пособие предназначено, по большей части, для студентов образовательного направления 38.03.05 «Бизнес-информатика». Однако, может быть полезным для всех интересующихся дисциплиной «Программирование».

УДК 681.3.05(075.8) + 681.3.068(075.8)

© В.В. Костерин, 2019

ОГЛАВЛЕНИЕ

Введение	8
Эко-среда для выполнения курсовой работы по программированию	8
Установка Python в Windows	8
Порядок установки.....	8
Проверка интерпретатора Python.....	12
Anaconda.....	14
Установка в Windows	14
Установка в Linux	18
Проверка работоспособности Anaconda.....	22
PyCharm.....	24
Установка PyCharm в Windows.....	24
Установка PyCharm в Linux	28
О PyCharm подробнее. Проверка работоспособности	29
Основные отличия Community Edition и Professional Edition	29
Ключевые возможности	30
Лицензирование.....	32
Создание и настройка проекта	32
Интерфейс и выполнение проекта	36
Отладка проекта	40
Разминка. 70 задач для обретения уверенности	45
Линейные алгоритмы.....	46
Ветвления и оператор выбора.....	46
Циклические алгоритмы. Обработка последовательностей и одномерных массивов (списков).....	48
Работа с ассоциативными массивами (таблицами данных)	52
Курсовая работа — часть 1. Экспериментальная оценка эффективности алгоритмов внутренней сортировки.....	54
Сортировка включением	55
Обменная сортировка	56

Сортировка выбором	56
Сравнение методов внутренней сортировки	57
Что надо сделать в первой части курсовой	58
Последовательность выполнения работы.....	61
Оформление отчёта по первой части	61
Вопросы для самоконтроля	62
Курсовая работа — часть 2: Матричная математика и работа с пакетами	63
<i>Кто такой этот NumPy?</i>	64
Настройка PyCharm и подключение библиотек NumPy и MathPlotLib	65
Примеры использования библиотек NumPy и Matplotlib	71
<i>Код примера «Визуализация функций»</i>	72
<i>Результат примера «Визуализация функций»</i>	73
<i>Код примера «Статистика»</i>	73
<i>Результат примера «Статистика»</i>	74
<i>Код примера «Линейная алгебра»</i>	75
<i>Результат примера «Линейная алгебра»</i>	76
Содержание части 2 курсовой работы	77
<i>Работа с двумерными массивами (матрицами)</i>	78
<i>Решение СЛАУ методом Гаусса — немного теории</i>	82
<i>Код программы решения СЛАУ методом Гаусса</i>	84
<i>Результат работы программы для решения СЛА методом Гаусса</i>	87
<i>Программа решения СЛАУ с использованием пакета numpy</i>	89
<i>Результат решения СЛАУ с использованием пакета numpy</i>	92
<i>Задание курсовой работы для решения СЛАУ методом Гаусса</i>	92
Библиографический список.....	92
Приложение 1.....	93
Шпаргалка Python №1	93
Приложение 2.....	94
Шпаргалка Python №2	94
Приложение 3.....	95

Приложение 4.....	96
Приложение 5.....	100
Варианты СЛАУ для курсовой работы.....	100
<i>Вариант №1</i>	<i>100</i>
<i>Вариант №2</i>	<i>101</i>
<i>Вариант №3</i>	<i>102</i>
<i>Вариант №4</i>	<i>103</i>
<i>Вариант №5</i>	<i>103</i>
<i>Вариант №6</i>	<i>104</i>
<i>Вариант №7</i>	<i>105</i>
<i>Вариант №8</i>	<i>106</i>
<i>Вариант №9</i>	<i>107</i>
<i>Вариант №10</i>	<i>108</i>
<i>Вариант №11</i>	<i>110</i>
<i>Вариант №12</i>	<i>111</i>
<i>Вариант №13</i>	<i>111</i>
<i>Вариант №14</i>	<i>112</i>
<i>Вариант №15</i>	<i>114</i>
<i>Вариант №16</i>	<i>114</i>
<i>Вариант №17</i>	<i>115</i>
<i>Вариант №18</i>	<i>116</i>
<i>Вариант №19</i>	<i>117</i>
<i>Вариант №20</i>	<i>118</i>
<i>Вариант №21</i>	<i>119</i>
<i>Вариант №22</i>	<i>120</i>
<i>Вариант №23</i>	<i>121</i>
<i>Вариант №24</i>	<i>122</i>
<i>Вариант №25</i>	<i>123</i>
<i>Вариант №26</i>	<i>124</i>
<i>Вариант №27</i>	<i>125</i>
<i>Вариант №28</i>	<i>126</i>
<i>Вариант №29</i>	<i>127</i>
<i>Вариант №30</i>	<i>127</i>

Введение

Привет, Пушкин...

Эко-среда для выполнения курсовой работы по программированию

На сегодняшний день существуют две версии Python — это Python 2 и Python 3, у них отсутствует полная совместимость друг с другом. На момент написания методических указаний вторая версия Python ещё используется, но, судя по изменениям, которые происходят, со временем, он останется только для того, чтобы запускать старый код. В нашей с вами работе, мы будем использовать Python 3, и, в дальнейшем, если где-то будет встречаться слово Python, то под ним следует понимать Python 3. Случаи применения Python 2 будут специально оговариваться.

Установка Python в Windows

Для установки интерпретатора Python на ваш компьютер, первое, что нужно сделать — это скачать дистрибутив. Загрузить его можно с [официального сайта](https://www.python.org/downloads/) <https://www.python.org/downloads/>.

Для операционной системы Windows дистрибутив распространяется либо в виде исполняемого файла (с расширением exe), либо в виде архивного файла (с расширением zip), который перед установкой необходимо распаковать.

Порядок установки

1. Запустите скачанный установочный файл.
2. Выберет способ установки.

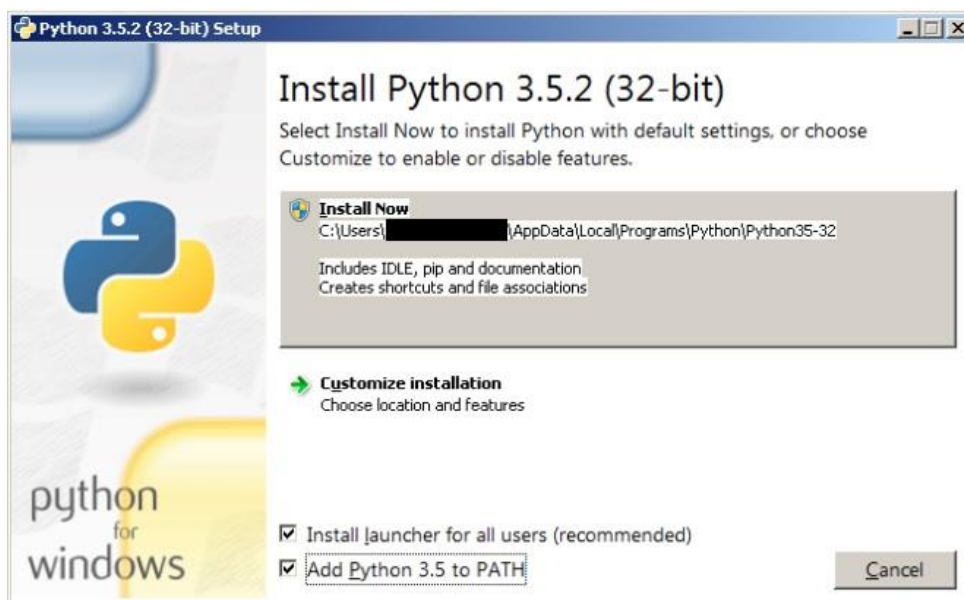


Рис. 1. Выбор способа установки Python

Здесь предлагается два варианта *Install Now* и *Customize installation*. При выборе *Install Now* Python установится в папку по указанному пути (Рис. 2). Помимо самого интерпретатора будет установлен IDLE (Integrated Development and Learning Environment — интегрированная среда разработки и обучения), pip (пакетный менеджер) и документация, а также будут созданы соответствующие ярлыки и установлены связи файлов, имеющие расширение.py с интерпретатором Python. *Customize installation* — это вариант настраиваемой установки. Опция Add python 3.5 to PATH нужна для того, чтобы появилась возможность запускать интерпретатор без указания полного пути до исполняемого файла при работе в командной строке.

3. Отметьте необходимые опций установки (доступно при выборе *Customize installation*).

На этом шаге нам предлагается отметить дополнения, устанавливаемые вместе с интерпретатором Python. Рекомендуем выбрать все опции (Рис. 3).

- **Documentation** — установка документаций.
- **pip** — установка пакетного менеджера pip.
- **tcl/tk and IDLE** — установка интегрированной среды разработки и обучения Python (IDLE) и библиотеки для построения графического интерфейса (tkinter).



Рис. 4. Настройки установки

4. Выберите место установки (доступно при выборе *Customize installation*)

Помимо указания пути, данное окно позволяет внести дополнительные изменения в процесс установки с помощью опций (Рис. 5):

- ***Install for all users*** — Установить для всех пользователей. Если не выбрать данную опцию, то будет предложен вариант инсталляции в папку пользователя, устанавливающего интерпретатор.
- ***Associate files with Python*** — Связать файлы, имеющие расширение.py, с Python. При выборе данной опции будут внесены изменения в настройки ассоциаций Windows, позволяющие запускать Python скрипты по двойному щелчку мыши.

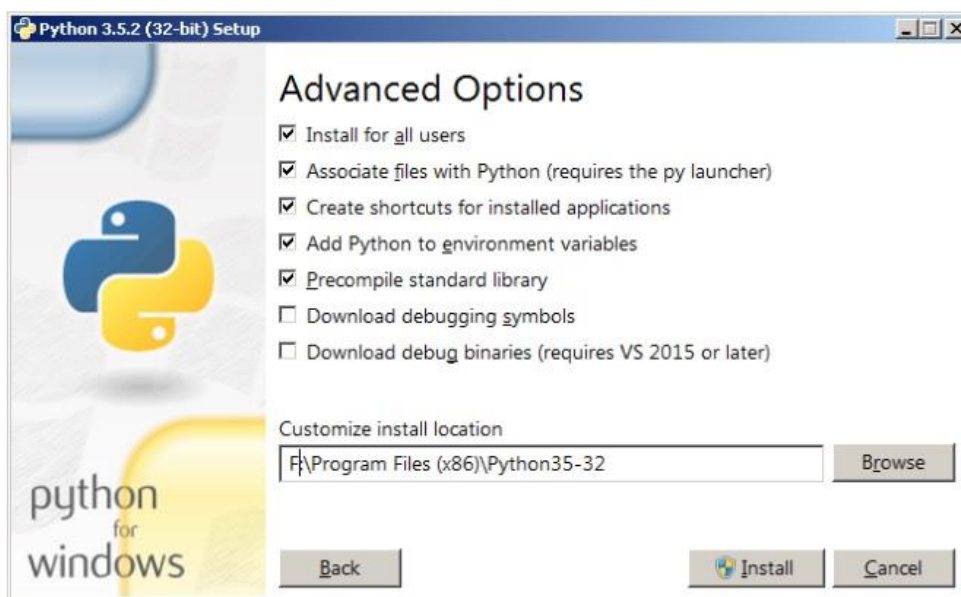


Рис. 6. Дополнительные настройки установки

- ***Create shortcuts for installed applications*** — Создать ярлыки для запуска приложений.
- ***Add Python to environment variables*** — Добавить пути до интерпретатора Python в переменную PATH.
- ***Precompile standard library*** — Провести предварительную компиляцию стандартной библиотеки.

Последние два пункта связаны с загрузкой компонентов для отладки, их мы устанавливать не будем.

5. После успешной установки вас ждет следующее сообщение — «Setup was Successful» (Рис. 7).



Рис. 8. Завершение установки Python

Проверка интерпретатора Python

Перед началом работы проверим интерпретатор в командном режиме. Если вы работаете в Windows, то нажмите сочетание Win+R и в появившемся окне введите python. В Linux откройте окно терминала и в нем введите python3 (или python).

В результате Python запустится в командном режиме. Выглядеть это будет примерно так, как на (картинка приведена для Windows, в Linux результат будет аналогичным) на Рис. 9 ниже:

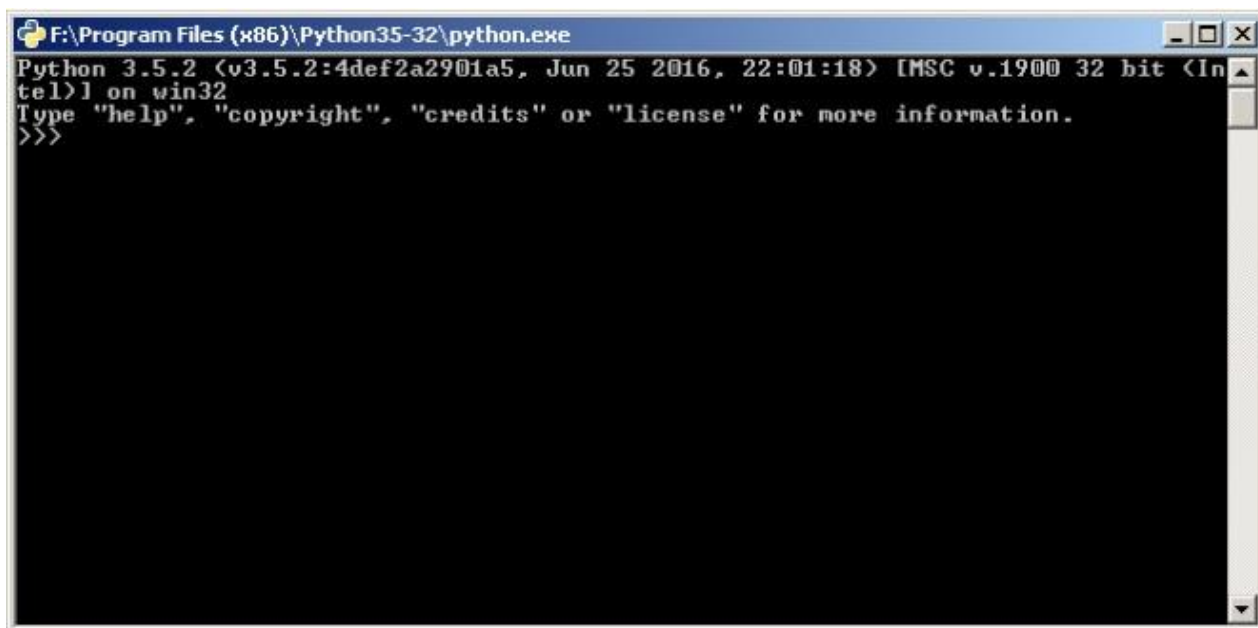


Рис. 10. Запуск интерпретатора Python

Если всё так, то в окне запишите `print("Hello, World!")` и нажмите <Enter> (Рис. 11)

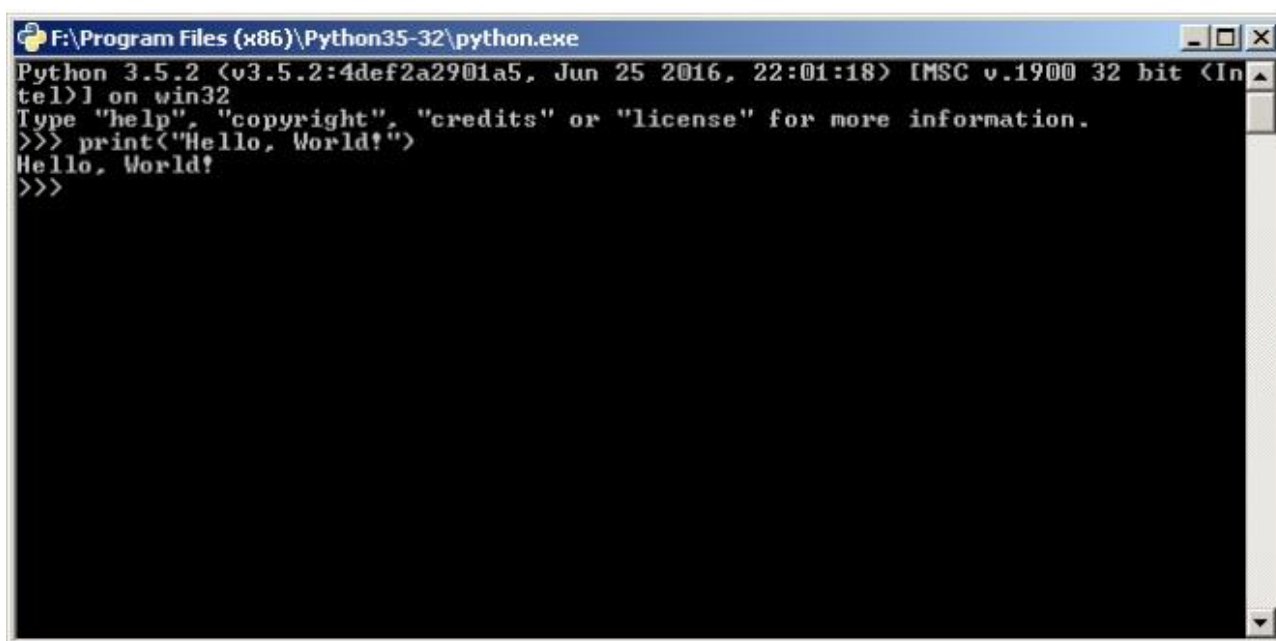


Рис. 12. В окне введите `print("Hello, World!")`

Anaconda

Кроме того, для удобства запуска примеров и изучения языка Python, советую установить на свой ПК пакет Anaconda. Этот пакет включает в себя интерпретатор языка Python (есть версии 2 и 3), набор наиболее часто используемых библиотек и удобную среду разработки и исполнения, запускаемую в браузере.

Для установки этого пакета, предварительно нужно скачать дистрибутив с сайта проекта <https://www.anaconda.com>.

Есть варианты под Windows, Linux и MacOS.

Установка в Windows

1. Запустите скачанный инсталлятор. В первом появившемся окне необходимо нажать «Next» (Рис. 13).



Рис. 14. Приглашение к установке Anaconda

2. Далее следует принять лицензионное соглашение (Рис. 15).

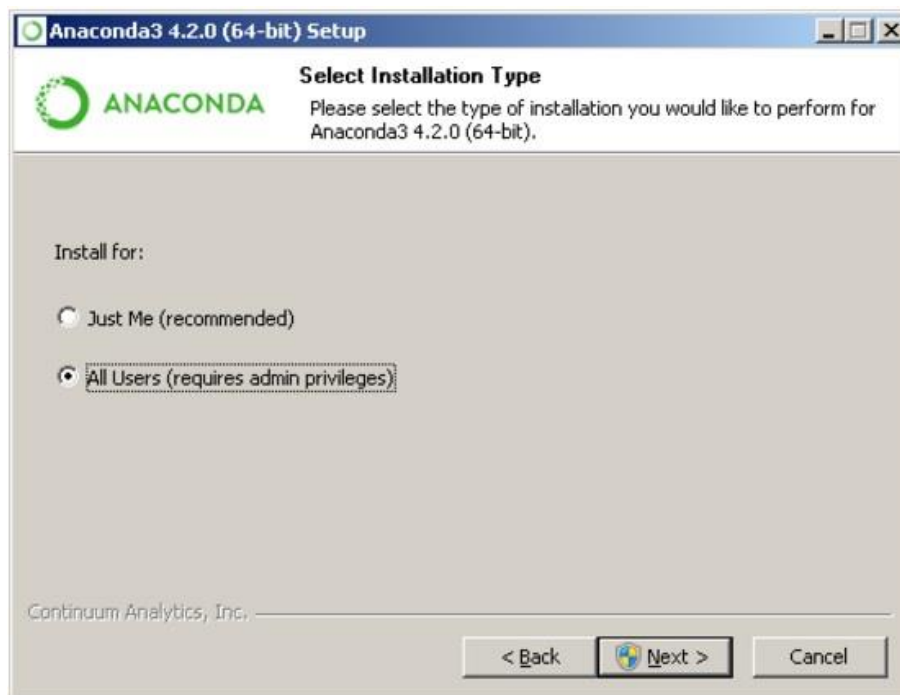


Рис. 18. Выбор разрешения для использования

4. Укажите путь, по которому будет установлена Anaconda (Рис. 19).

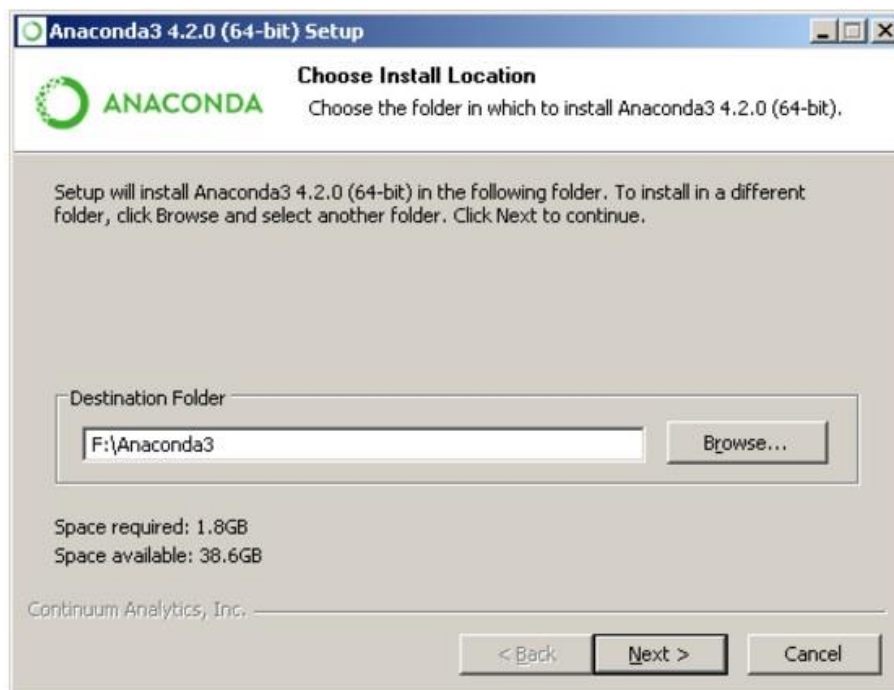


Рис. 20. Настройка пути установки

5. Укажите дополнительные опции (Рис. 21):

Add Anaconda to the system PATH environment variable — добавить Anaconda в системную переменную PATH.

- *Register Anaconda as the system Python 3.5* — использовать Anaconda, как интерпретатор Python 3.5 по умолчанию.
- Для начала установки нажмите на кнопку «*Install*».

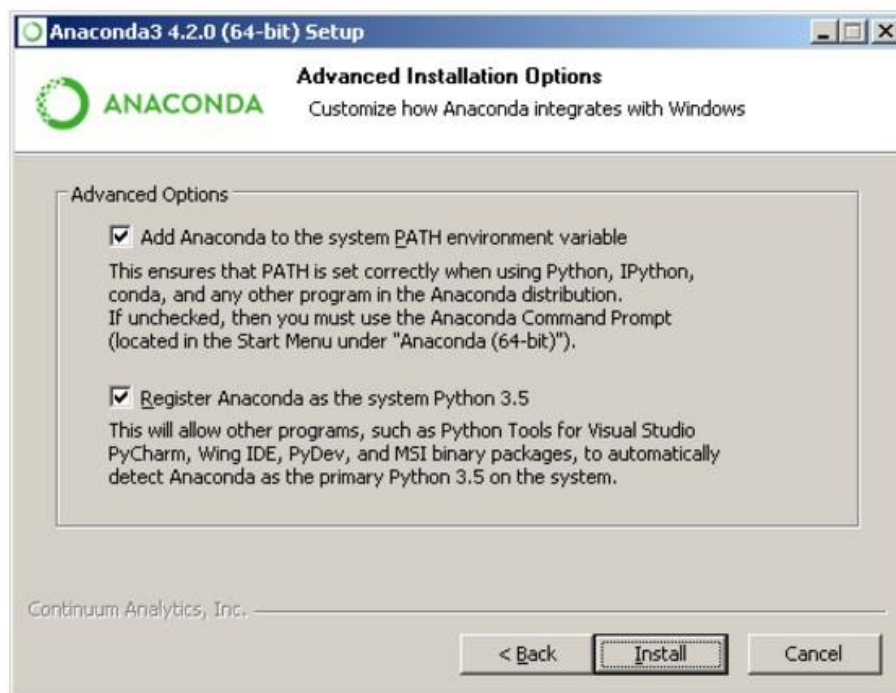


Рис. 22. Дополнительные настройки установки Anaconda

5. После этого будет произведена установка Anaconda на ваш компьютер (Рис. 23).

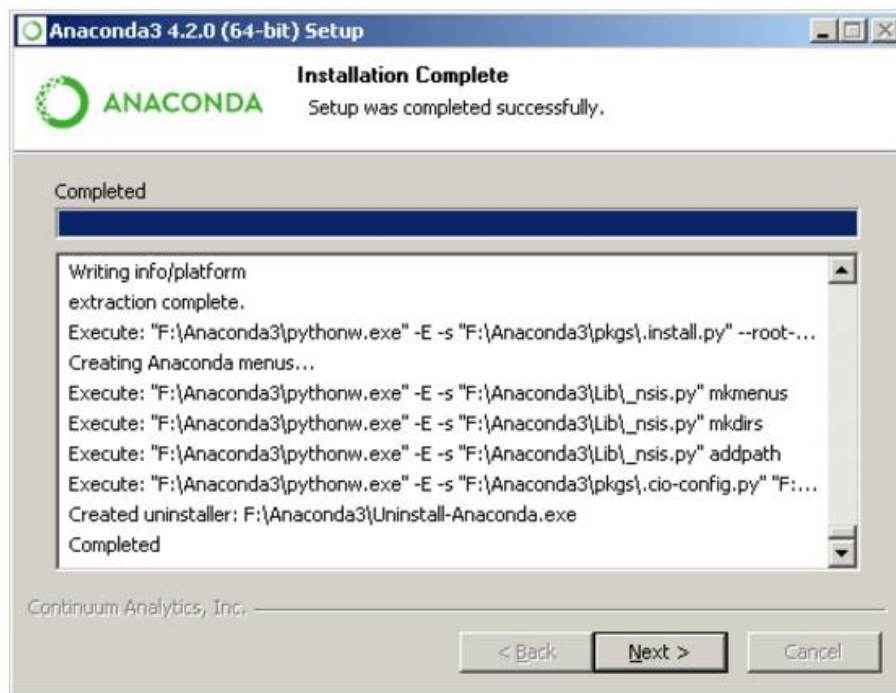


Рис. 24. Индикатор процесса установки

Установка в Linux

Скачайте дистрибутив Anaconda для Linux, он будет иметь расширение.sh, и запустите установку командой:

```
bash имя_дистрибутива.sh
```

В результате вы увидите приглашение к установке. Для продолжения нажмите «Enter» (Рис. 25).

```
Terminal - tester@tester-VirtualBox: ~/Downloads
File Edit View Terminal Tabs Help
WARNING: The "syslog" option is deprecated
Enter tester's password:
session setup failed: NT_STATUS_LOGON_FAILURE
tester@tester-VirtualBox:/etc/samba$ cd ~
tester@tester-VirtualBox:~$ ls
code  dist  Downloads  Music  Public  Templates  vm
Desktop  Documents  lab  Pictures  temp  Videos  winetest
tester@tester-VirtualBox:~$ cd Do
bash: cd: Do: No such file or directory
tester@tester-VirtualBox:~$ cd Downloads/
tester@tester-VirtualBox:~/Downloads$ ls
Anaconda3-4.2.0-Linux-x86_64.sh  vim-colors-solarized-master.zip
codeschool-vim-theme-master    vim-distinguished-develop
codeschool-vim-theme-master.zip vim-distinguished-develop.zip
molokai.vim                    Vistaluna_Basic_by_Pgase.rar
vim-colors-solarized-master
tester@tester-VirtualBox:~/Downloads$ bash Anaconda3-4.2.0-Linux-x86_64.sh

Welcome to Anaconda3 4.2.0 (by Continuum Analytics, Inc.)

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>> |
```

Рис. 26. Начало установки Anaconda

2. Прочитайте лицензионное соглашение, его нужно пролистать до конца (Рис. 27).

```
Terminal - tester@tester-VirtualBox: ~/Downloads
File Edit View Terminal Tabs Help
=====
Anaconda License
=====

Copyright 2016, Continuum Analytics, Inc.

All rights reserved under the 3-clause BSD License:

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

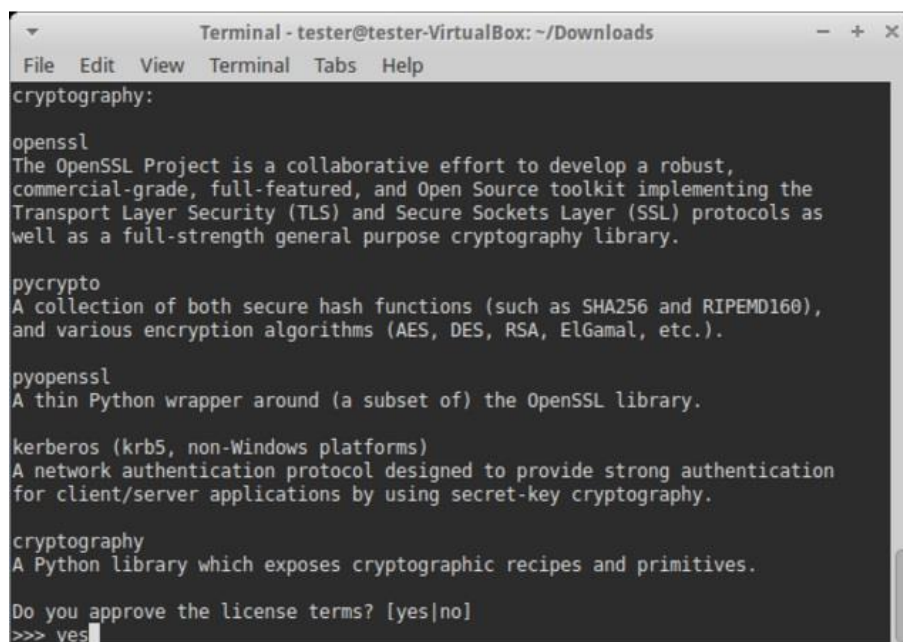
* Neither the name of Continuum Analytics, Inc. nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
--More--
```

Рис. 28. Лицензионное соглашение, его нужно пролистать до конца

Согласитесь с ним, для этого требуется набрать в командной строке «yes», в ответе на вопрос инсталлятора (Рис. 29):

Do you approve the license terms? [yes/no]



```
Terminal - tester@tester-VirtualBox: ~/Downloads
File Edit View Terminal Tabs Help
cryptography:
openssl
The OpenSSL Project is a collaborative effort to develop a robust,
commercial-grade, full-featured, and Open Source toolkit implementing the
Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols as
well as a full-strength general purpose cryptography library.

pycrypto
A collection of both secure hash functions (such as SHA256 and RIPEMD160),
and various encryption algorithms (AES, DES, RSA, ElGamal, etc.).

pyopenssl
A thin Python wrapper around (a subset of) the OpenSSL library.

kerberos (krb5, non-Windows platforms)
A network authentication protocol designed to provide strong authentication
for client/server applications by using secret-key cryptography.

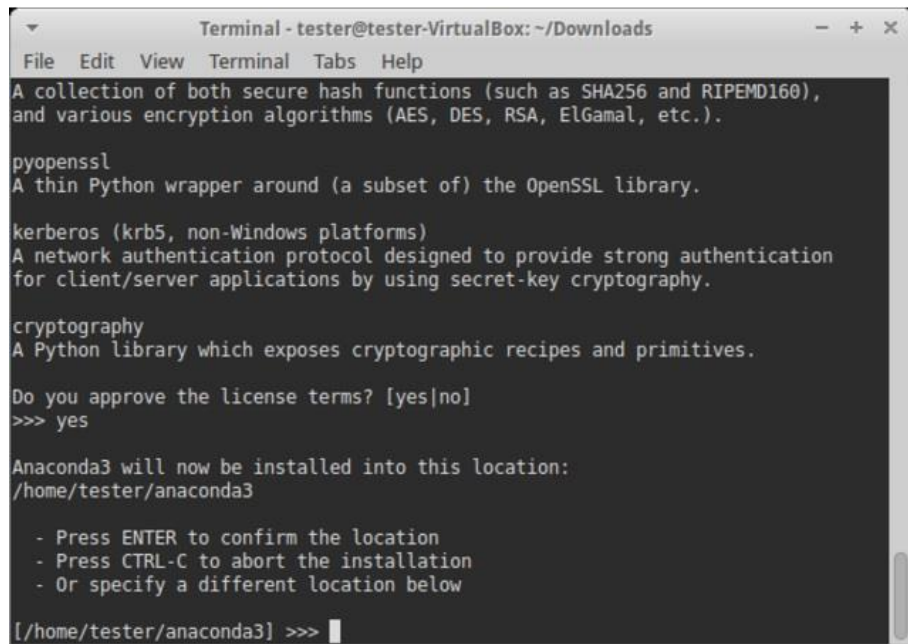
cryptography
A Python library which exposes cryptographic recipes and primitives.

Do you approve the license terms? [yes/no]
>>> yes
```

Рис. 30. Соглашайтесь с лицензионным соглашением

3. Выберите место установки. Можно выбрать один из следующих вариантов (Рис. 31):

- *Press ENTER to confirm the location* — для моей машины предложен путь: `/home/tester/anaconda3`, нажимаю «Enter».
- *Press CTRL-C to abort the installation* — CTRL-C для отмены установки.
- *Or specify a different location below* — другой путь в строке ниже.

A terminal window titled "Terminal - tester@tester-VirtualBox: ~/Downloads" showing the Anaconda3 installation process. The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The output text includes descriptions of various libraries: "A collection of both secure hash functions (such as SHA256 and RIPEMD160), and various encryption algorithms (AES, DES, RSA, ElGamal, etc.).", "pyopenssl: A thin Python wrapper around (a subset of) the OpenSSL library.", "kerberos (krb5, non-Windows platforms): A network authentication protocol designed to provide strong authentication for client/server applications by using secret-key cryptography.", and "cryptography: A Python library which exposes cryptographic recipes and primitives." It then asks "Do you approve the license terms? [yes|no]" and receives the input ">>> yes". Next, it states "Anaconda3 will now be installed into this location: /home/tester/anaconda3" and lists instructions: "- Press ENTER to confirm the location", "- Press CTRL-C to abort the installation", and "- Or specify a different location below". The prompt is now "[/home/tester/anaconda3] >>>".

```
Terminal - tester@tester-VirtualBox: ~/Downloads
File Edit View Terminal Tabs Help
A collection of both secure hash functions (such as SHA256 and RIPEMD160),
and various encryption algorithms (AES, DES, RSA, ElGamal, etc.).

pyopenssl
A thin Python wrapper around (a subset of) the OpenSSL library.

kerberos (krb5, non-Windows platforms)
A network authentication protocol designed to provide strong authentication
for client/server applications by using secret-key cryptography.

cryptography
A Python library which exposes cryptographic recipes and primitives.

Do you approve the license terms? [yes|no]
>>> yes

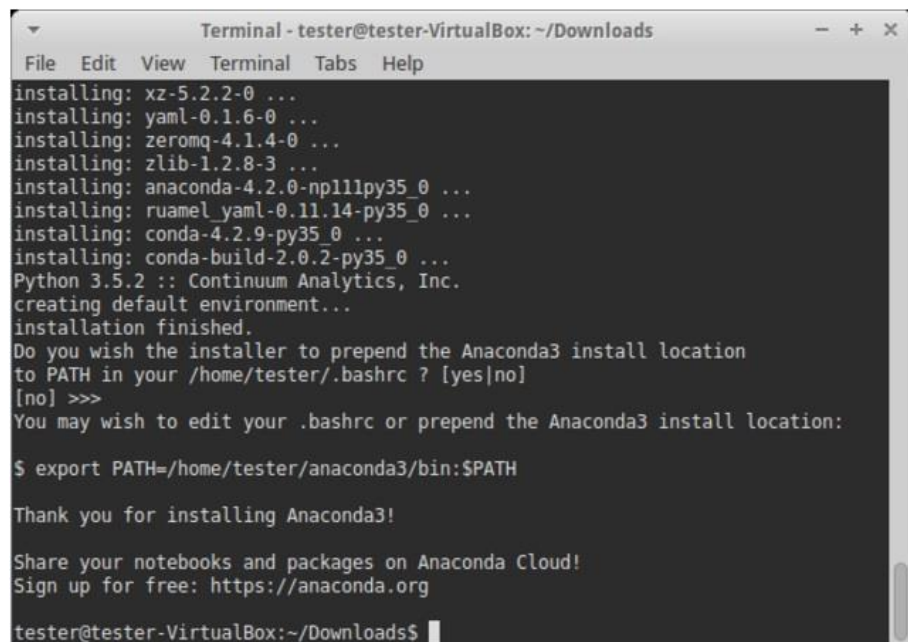
Anaconda3 will now be installed into this location:
/home/tester/anaconda3

- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below

[/home/tester/anaconda3] >>>
```

Рис. 32. Выбираем место установки

4. После этого начнется установка (Рис. 33).

A terminal window titled "Terminal - tester@tester-VirtualBox: ~/Downloads" showing the final steps of Anaconda3 installation. It lists several packages being installed: "xz-5.2.2-0 ...", "yaml-0.1.6-0 ...", "zeromq-4.1.4-0 ...", "zlib-1.2.8-3 ...", "anaconda-4.2.0-np111py35_0 ...", "ruamel_yaml-0.11.14-py35_0 ...", "conda-4.2.9-py35_0 ...", and "conda-build-2.0.2-py35_0 ...". It then states "Python 3.5.2 :: Continuum Analytics, Inc." and "creating default environment...". The installation is finished, and it asks "Do you wish the installer to prepend the Anaconda3 install location to PATH in your /home/tester/.bashrc ? [yes|no]" with the input "[no] >>>". It then suggests "You may wish to edit your .bashrc or prepend the Anaconda3 install location:" and shows the command "\$ export PATH=/home/tester/anaconda3/bin:\$PATH". It concludes with "Thank you for installing Anaconda3!", a link to "Share your notebooks and packages on Anaconda Cloud!", and a sign-up link "Sign up for free: https://anaconda.org". The prompt is now "tester@tester-VirtualBox:~/Downloads\$".

```
Terminal - tester@tester-VirtualBox: ~/Downloads
File Edit View Terminal Tabs Help
installing: xz-5.2.2-0 ...
installing: yaml-0.1.6-0 ...
installing: zeromq-4.1.4-0 ...
installing: zlib-1.2.8-3 ...
installing: anaconda-4.2.0-np111py35_0 ...
installing: ruamel_yaml-0.11.14-py35_0 ...
installing: conda-4.2.9-py35_0 ...
installing: conda-build-2.0.2-py35_0 ...
Python 3.5.2 :: Continuum Analytics, Inc.
creating default environment...
installation finished.
Do you wish the installer to prepend the Anaconda3 install location
to PATH in your /home/tester/.bashrc ? [yes|no]
[no] >>>
You may wish to edit your .bashrc or prepend the Anaconda3 install location:

$ export PATH=/home/tester/anaconda3/bin:$PATH

Thank you for installing Anaconda3!

Share your notebooks and packages on Anaconda Cloud!
Sign up for free: https://anaconda.org

tester@tester-VirtualBox:~/Downloads$
```

Рис. 34. Процесс установки

Проверка работоспособности Anaconda

Теперь проверим работоспособность всего того, что мы установили.

Здесь и далее будем считать, что пакет Anaconda установлен в Windows, в папку C:\Anaconda3, в Linux, вы его найдёте в каталоге, который выбрали при установке.

Перейдите в папку `Scripts` и введите в командной строке:

ipython notebook

Если вы находитесь в Windows и открыли папку C:\Anaconda3\Scripts через проводник, то для запуска интерпретатора командной строки для этой папки в поле адреса введите cmd (Рис. 35).

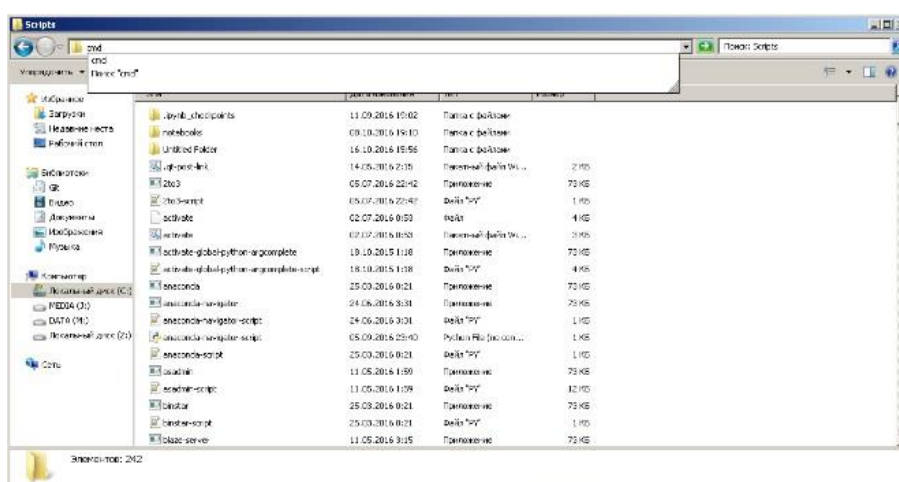
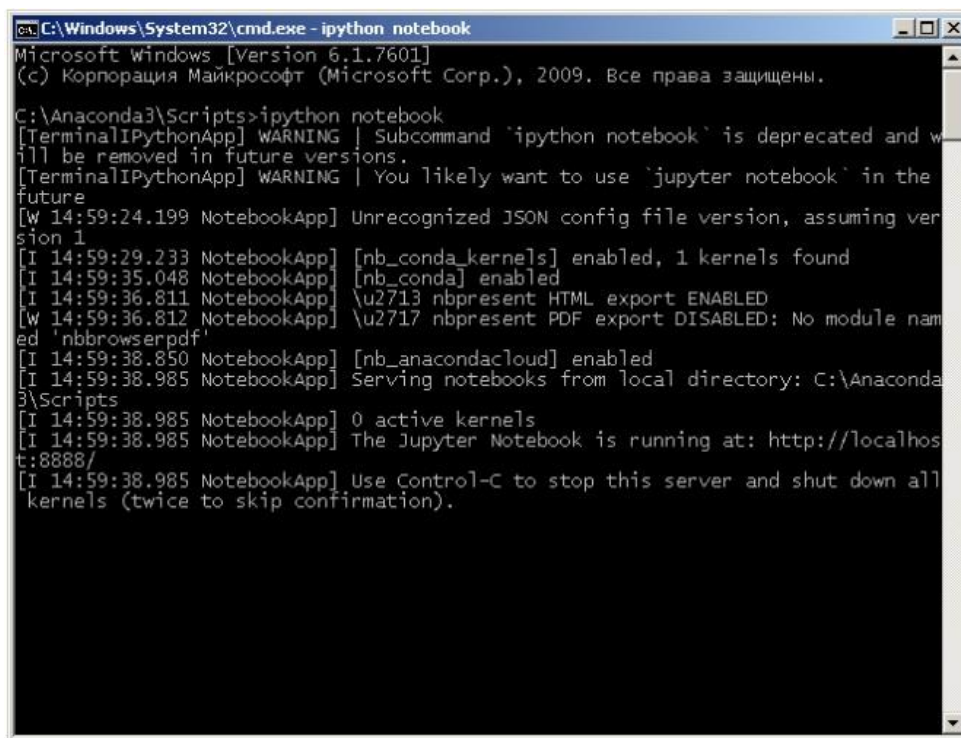


Рис. 36. Запускаем интерпретатор



```
C:\Windows\System32\cmd.exe - ipython notebook
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Anaconda3\Scripts>ipython notebook
[TerminalIPythonApp] WARNING | Subcommand 'ipython notebook' is deprecated and will be removed in future versions.
[TerminalIPythonApp] WARNING | You likely want to use 'jupyter notebook' in the future
[W 14:59:24.199 NotebookApp] Unrecognized JSON config file version, assuming version 1
[I 14:59:29.233 NotebookApp] [nb_conda_kernels] enabled, 1 kernels found
[I 14:59:35.048 NotebookApp] [nb_conda] enabled
[I 14:59:36.811 NotebookApp] \u2713 nbpresent HTML export ENABLED
[W 14:59:36.812 NotebookApp] \u2717 nbpresent PDF export DISABLED: No module named 'nbbrowserpdf'
[I 14:59:38.850 NotebookApp] [nb_anacondacloud] enabled
[I 14:59:38.985 NotebookApp] Serving notebooks from local directory: C:\Anaconda3\Scripts
[I 14:59:38.985 NotebookApp] 0 active kernels
[I 14:59:38.985 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 14:59:38.985 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

Рис. 37. Сейчас запустим Jupyter

В результате запустится веб-сервер и среда разработки в браузере (Рис. 38-39).

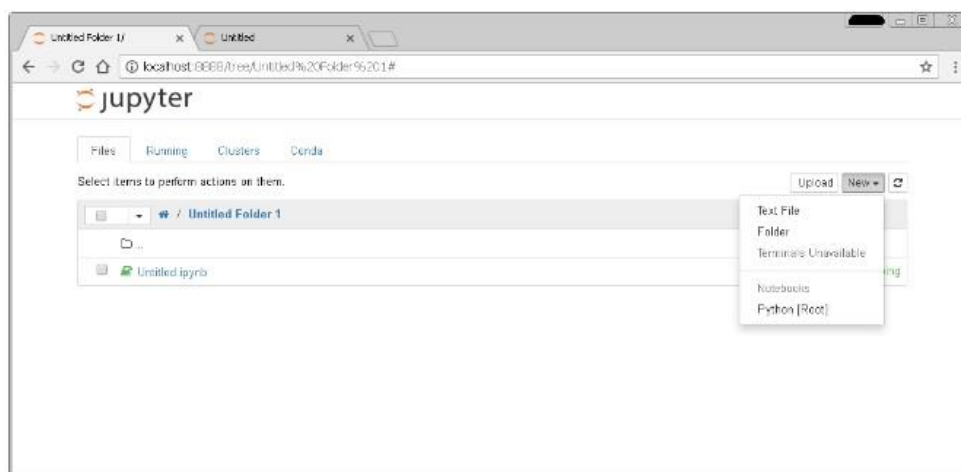


Рис. 40. Jupyter заработал в браузере

Создайте ноутбук для разработки, для этого нажмите на кнопку *New* (в правом углу окна) и в появившемся списке выберете *Python*.

В результате будет создана новая страница в браузере с ноутбуком. Введите в первой ячейке команду


```
print("Hello, World!")
```

и нажмите *Alt+Enter* на клавиатуре. Ниже ячейки должна появиться соответствующая надпись (Рис. 41).

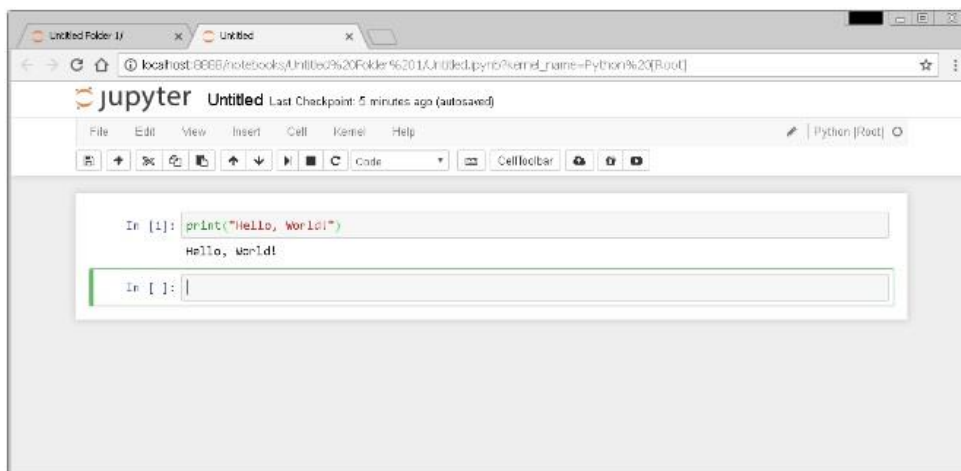


Рис. 42. Результат проверки Jupiter

PyCharm

Если в процессе разработки вам необходим отладчик и вообще вы привыкли работать в IDE, Integrated Development Environment — интегрированная среда разработки, то тогда одним из лучших вариантов будет IDE PyCharm от JetBrains. Для скачивания данного продукта нужно перейти по ссылке <https://www.jetbrains.com/pycharm/download/>

IDE доступна для Windows, Linux и MacOS. Существуют два вида лицензии PyCharm — это Professional и Community. Мы будем использовать версию Community, так как она бесплатна и её функционала более чем достаточно для наших задач¹.

Установка PyCharm в Windows

1. Запустите скачанный дистрибутив PyCharm (Рис. 43).

¹ Компания JetBrains любезно предоставила студентам образовательного направления 38.03.05 «Бизнес-информатика» Южно-Уральского государственного университета лицензии на все свои продукты в версии Professional бесплатно до конца обучения



Рис. 44. Приглашение к установке PyCharm

2. Выберите путь установки программы (Рис. 45).

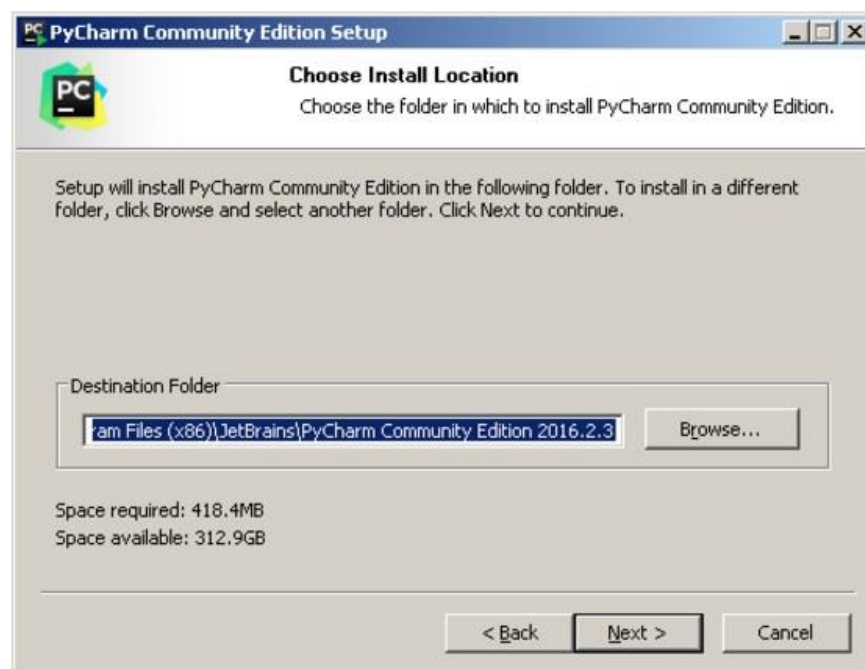


Рис. 46. Выбираем путь для установки

3. Выбор версии PyCharm, ассоциация файлов с расширением.py с PyCharm (Рис. 47).

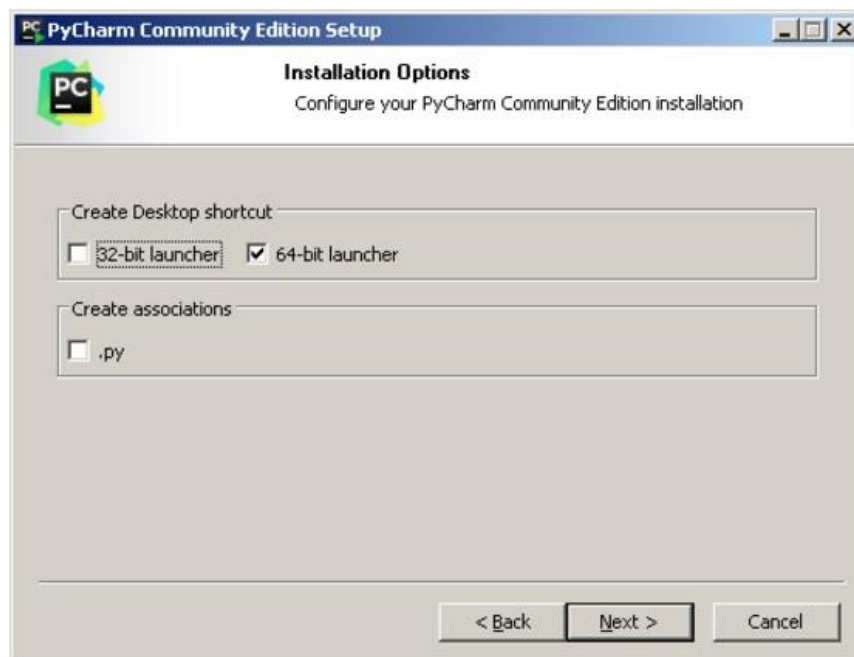


Рис. 48. Настройка версии и ассоциации файлов.py

4. Выберите имя для папки в меню Пуск. Завершение установки (Рис. 49-50):



Рис. 51. Выбор стартовой папки в меню

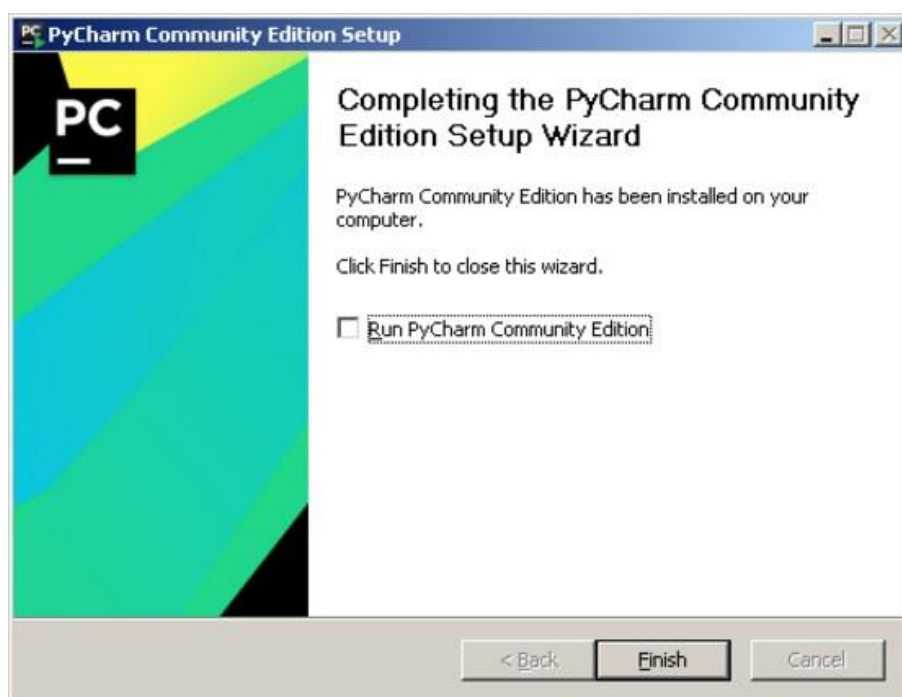


Рис. 52. Завершение установки

Установка PyCharm в Linux

1. Скачайте с сайта дистрибутив на компьютер.
2. Распакуйте архивный файл, для этого можно воспользоваться командой (Рис. 53):

```
> tar xvf имя_архива.tar.gz
```

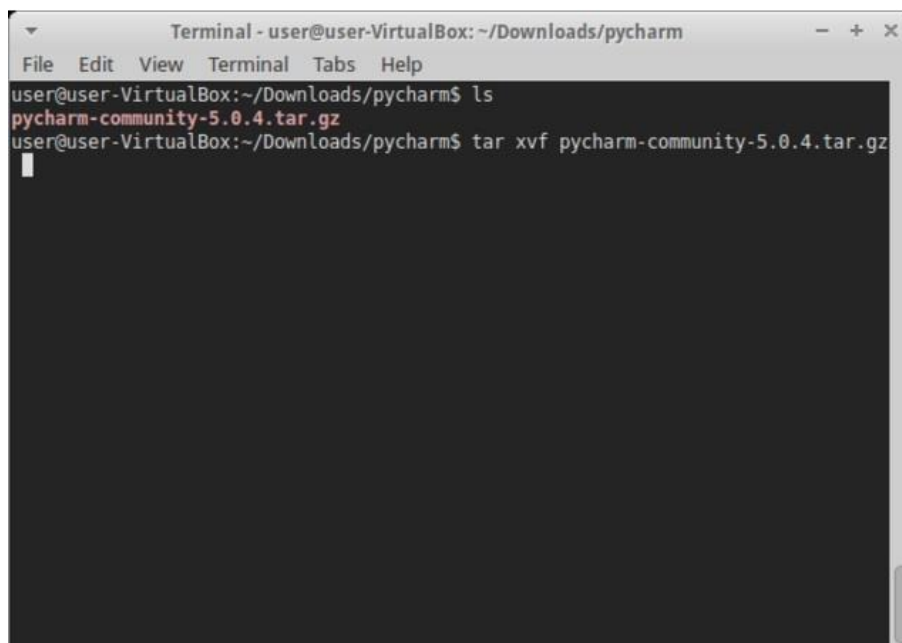
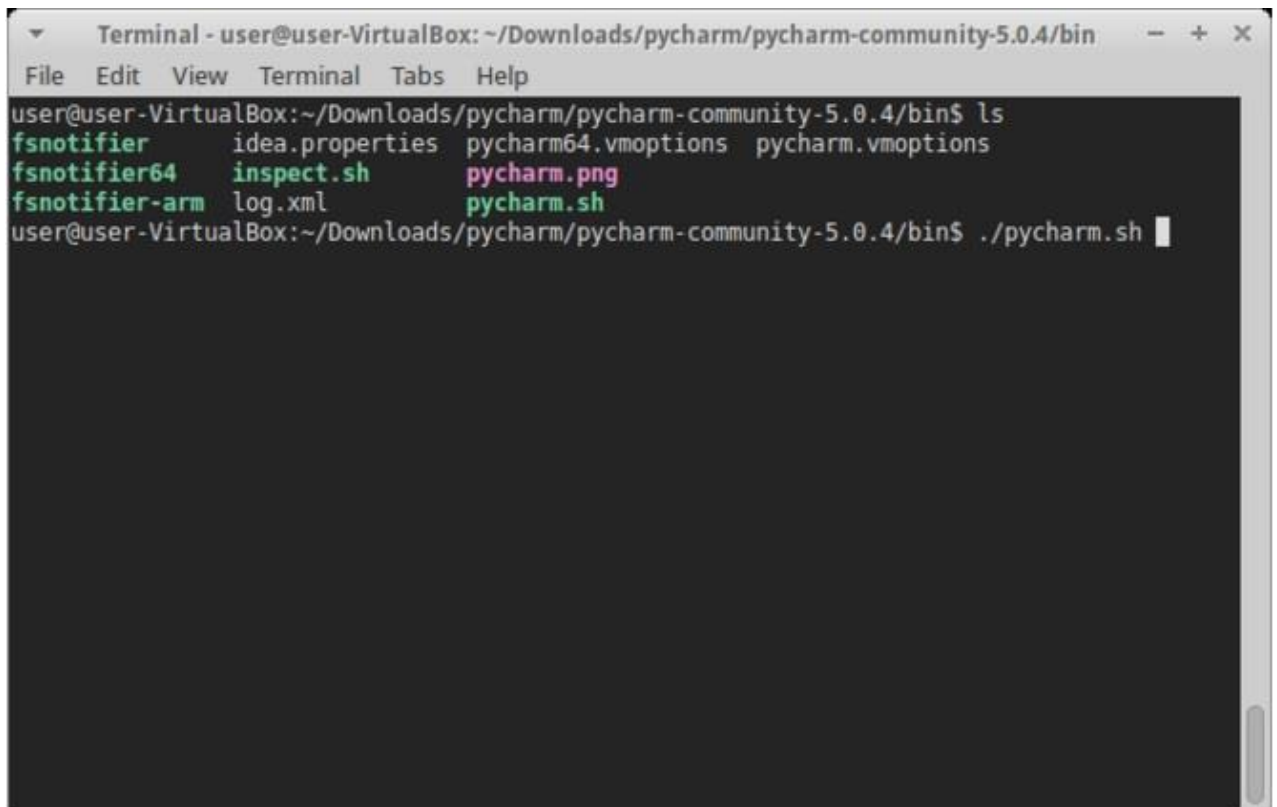


Рис. 54. Распакуем дистрибутив

Перейдите в каталог, который был создан после распаковки дистрибутива, найдите в нем подкаталог `bin` и зайдите в него. Запустите `pycharm.sh` командой (Рис. 55):



```
Terminal - user@user-VirtualBox: ~/Downloads/pycharm/pycharm-community-5.0.4/bin
File Edit View Terminal Tabs Help
user@user-VirtualBox:~/Downloads/pycharm/pycharm-community-5.0.4/bin$ ls
fsnotifier      idea.properties  pycharm64.vmoptions  pycharm.vmoptions
fsnotifier64    inspect.sh        pycharm.png
fsnotifier-arm  log.xml          pycharm.sh
user@user-VirtualBox:~/Downloads/pycharm/pycharm-community-5.0.4/bin$ ./pycharm.sh
```

Рис. 56. Переходим в нужный каталог и запускаем установку

В результате должен запускаться PyCharm.

О PyCharm подробнее. Проверка работоспособности

PyCharm — это smart IDE с полным набором средств эффективной разработки на языке Python. Выпускается в двух вариантах — бесплатная версия **PyCharm Community Edition** и, поддерживающая максимальный набор возможностей, **PyCharm Professional Edition**. PyCharm выполняет инспекцию кода на лету, автодополнение, в том числе основываясь на информации, полученной во время исполнения кода, навигацию по коду, обеспечивает множество рефакторингов.

Основные отличия Community Edition и Professional Edition

Community edition:

- Облегчённая IDE для разработки только на Python
- Бесплатная, с открытым кодом, под лицензией Apache 2

- Понимающий контекст редактор, отладчик, рефакторинги, инспекции, интеграция с VCS
- Навигация по проекту, поддержка тестирования, настраиваемый UI², горячие клавиши Vim

Professional Edition:

- Полнофункциональная IDE для разработки на Python, в том числе для многоязычных веб-приложений с фреймворками
- Поддержка фреймворков Django, Flask, Google App Engine, Pyramid, web2py
- Поддержка языков JavaScript, CoffeeScript, TypeScript, CSS, Cython и др.
- Удаленная разработка, Поддержка работы с БД и языка SQL
- Обнаружение дублирующегося кода
- Диаграммы UML & SQLAlchemy
- Python Profiler

Ключевые возможности

- Мощный и функциональный редактор кода с подсветкой синтаксиса, авто-форматированием и авто-отступами для поддерживаемых языков.
- Простая и мощная навигация в коде.
- Помощь при написании кода, включающая в себя авто-дополнение, авто-импорт, шаблоны кода, проверка на совместимость версии интерпретатора языка, и многое другое.

² https://ru.wikipedia.org/wiki/Интерфейс_пользователя

- Быстрый просмотр документации для любого элемента прямо в окне редактора, просмотр внешней документации через браузер, поддержка docstring — генерация, подсветка, авто-дополнение и многое другое.
- Большое количество инспекций кода.
- Мощный рефакторинг кода, который предоставляет широкие возможности по выполнению быстрых глобальных изменений в проекте.
- Полная поддержка свежих версий Django фреймворка.
- Поддержка Google App Engine.
- Поддержка IronPython, Jython, Cython, PyPy, wxPython, PyQt, PyGTK и др.
- Редактор Javascript, Coffescript, HTML/CSS, SASS, LESS, HAML.
- Интеграция с системами контроля версий (VCS).
- UML диаграммы классов, диаграммы моделей Django и Google App Engine.
- Интегрированное Unit тестирование.
- Интерактивные консоли для Python, Django, SSH, отладчика и баз данных.
- Полнофункциональный графический отладчик (Debugger).
- Поддержка схем наиболее популярных IDE/редакторов. таких как Netbeans, Eclipse, Emacs, эмуляция VIM редактора.
- Поддерживаемые языки: Python (V:2.x, 3.x), Jython, Cython, IronPython, PyPy, Javascript, CoffeScript, HTML/CSS, Django/Jinja2 templates, Gql, LESS/SASS/SCSS/HAML, Mako, Puppet, RegExp, Rest, SQL, XML, YAML.
- PyCharm имеет несколько цветовых схем, а также настраиваемую подсветку синтаксиса кода.

- Интеграция с баг/issue-трекерами JIRA, Youtrack, Lighthouse, Pivotal Tracker, GitHub, Redmine, Trac...
- Огромная, постоянно пополняемая коллекция плагинов.
- Кросс-платформенность (Windows, Mac OS X, Linux).

Лицензирование

PyCharm Professional Edition имеет несколько вариантов лицензий, которые отличаются функциональностью, стоимостью и условиями использования. Является бесплатным для образовательных учреждений и проектов с открытым исходным кодом.

PyCharm Community Edition — бесплатная версия, обладающая усеченным набором возможностей. Распространяется под лицензией Apache 2.

Таблица 1. Системные требования

Windows	OS X	Linux
Microsoft Windows 10/8/7/Vista/2003/XP (incl.64-bit)	Mac OS X 10.8 or higher	Linux, Ubuntu, Debian
1 GB RAM minimum, 2 GB RAM recommended	1 GB RAM minimum, 2 GB RAM recommended	512 MB RAM minimum, 1 GB RAM recommended
1024x768 minimum screen resolution	1024x768 minimum screen resolution	1024x768 minimum screen resolution
Python 2.4 or higher, Jython, PyPy or IronPython	Python 2.4 or higher, Jython, PyPy or IronPython	Python 2.4 or higher, Jython, PyPy or IronPython

Создание и настройка проекта

Все процессы описаны для версии PyCharm Professional 2016.1.4 (см. Рис. 57).



Рис. 58. PyCharm загружается

Для создания нового проекта в меню File — выбрать пункт New Project (Рис. 59).

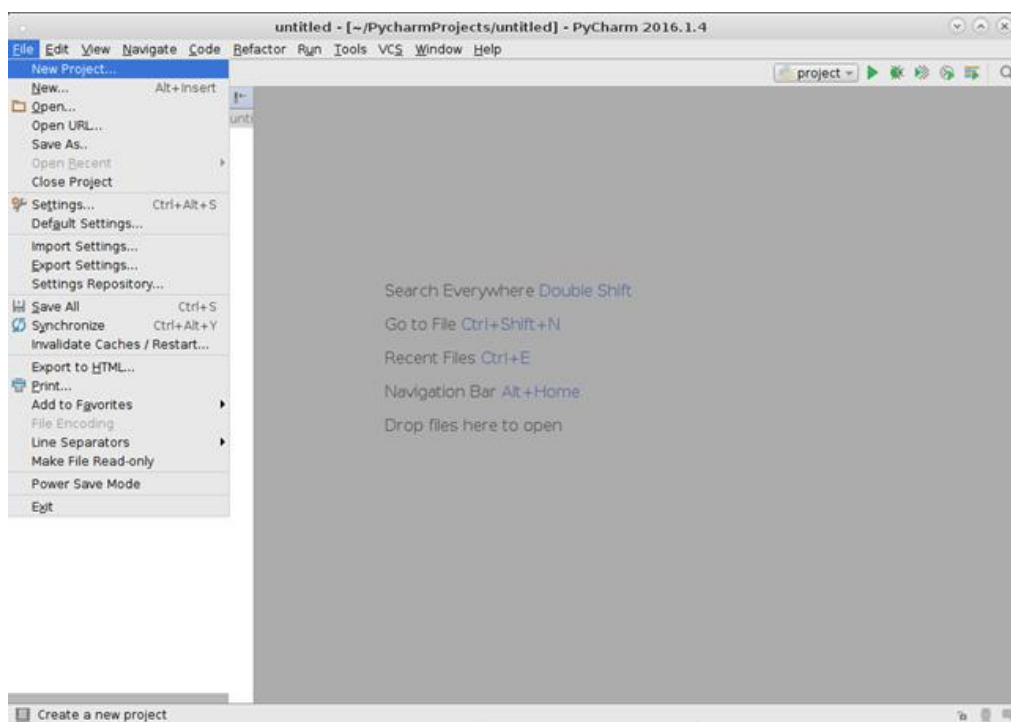


Рис. 60. Создание проекта

В появившемся окне указываем параметры создаваемого проекта. В поле Location можно указать путь, по которому будет храниться создаваемый проект. В поле Interpreter выбираем версию Python под которой необходимо создать проект. В данном примере выбрана версия Python 2.7 (Рис. 61).

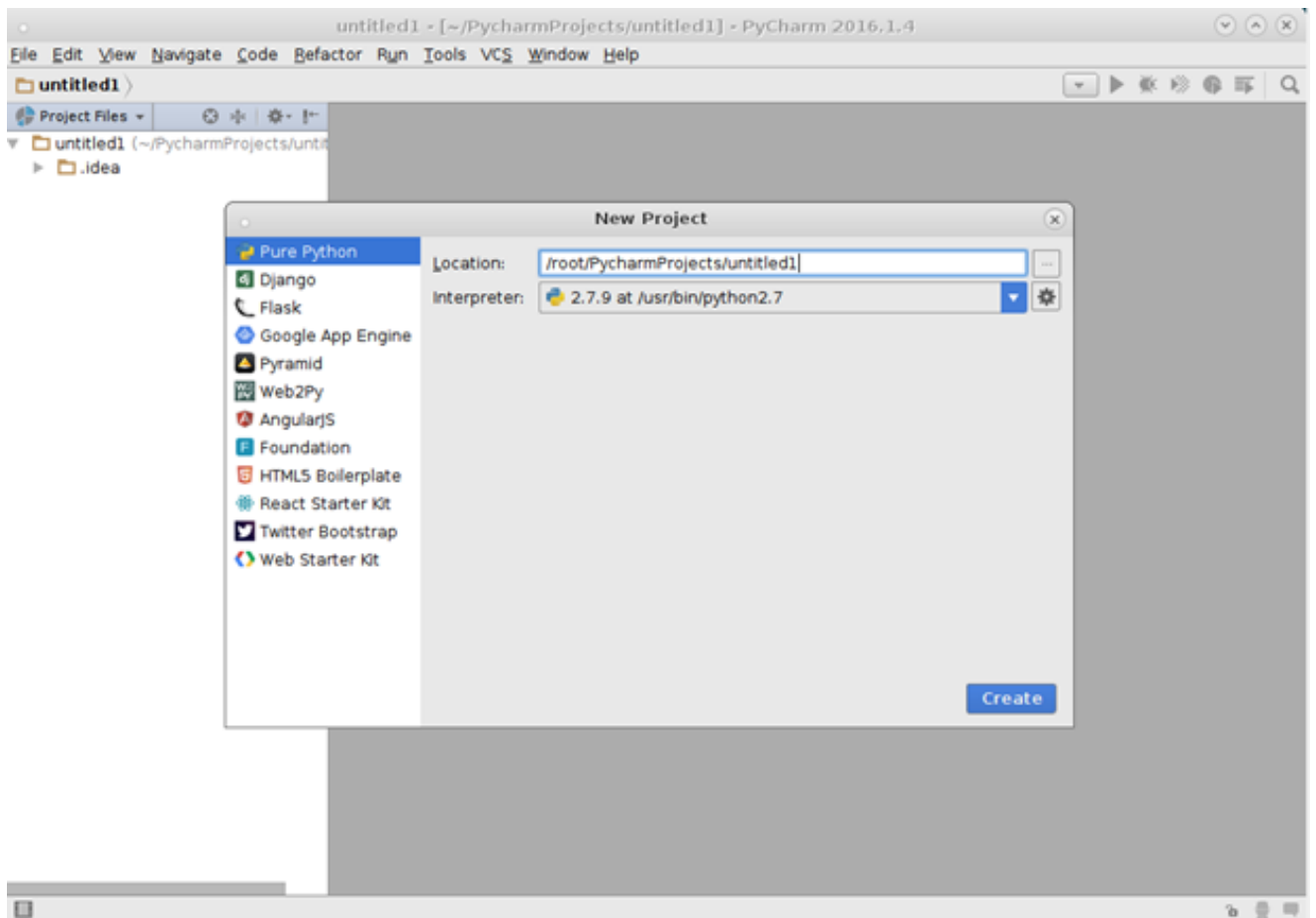


Рис. 62. Задаём имя проекта и версию интерпретатора

Для завершения создания проекта нажать кнопку *Create*. Можно выбрать, как создастся новый проект — в новом окне или в текущем (Рис. 63).

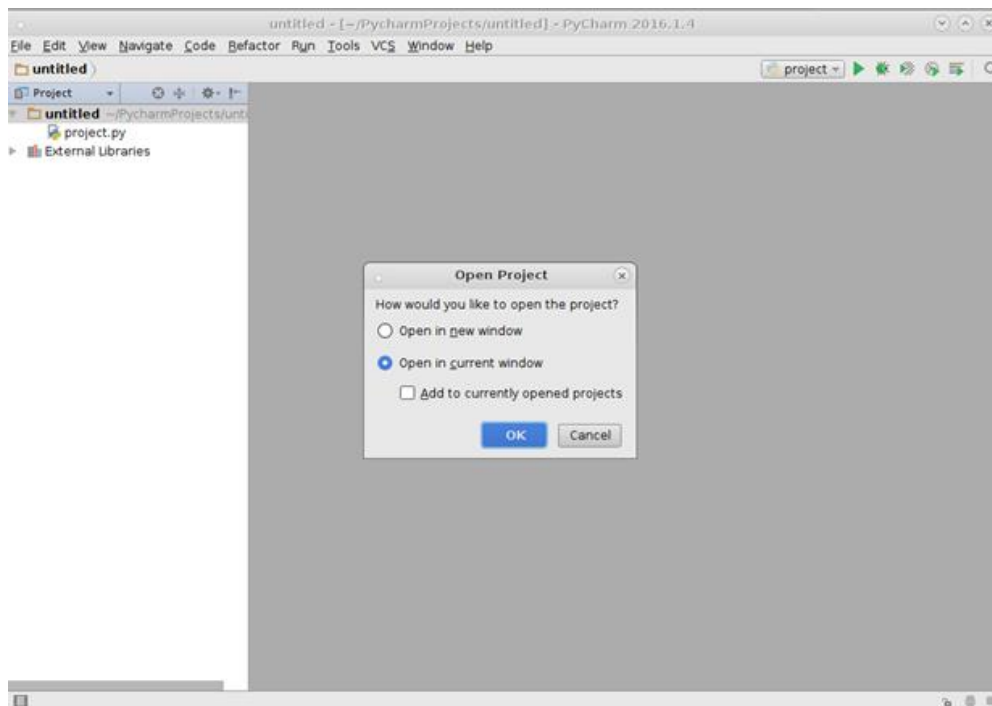


Рис. 64. Создаём новый проект в новом окне или в текущем

Проект создан, пора создать файл с расширением.py. Делает правой кнопкой «мыши» *New*→*Python File* (Рис. 65-66).

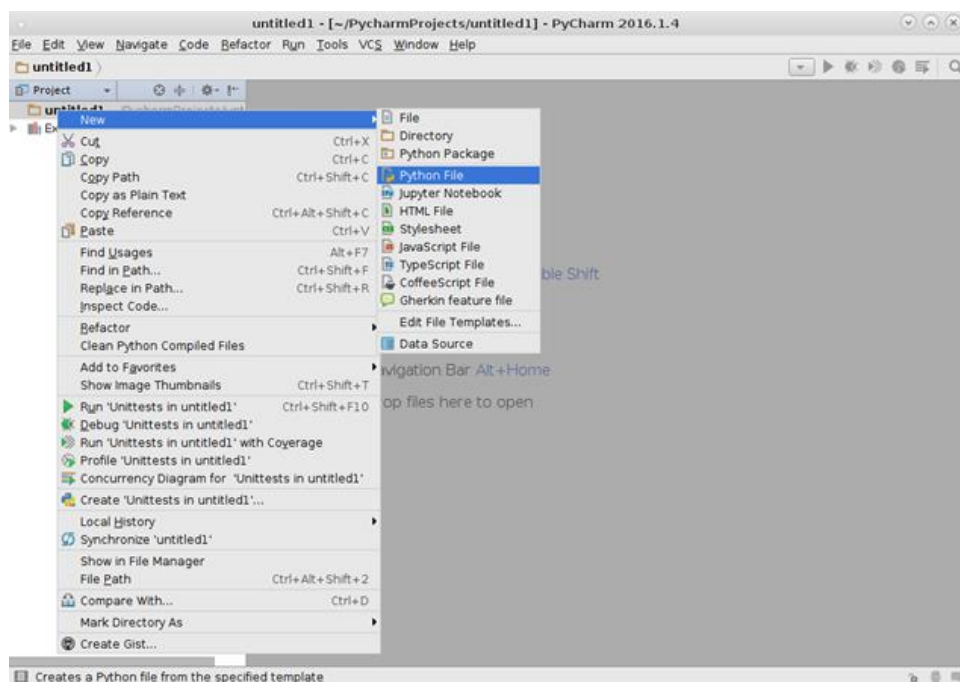


Рис. 67. Создаем файл для записи программы

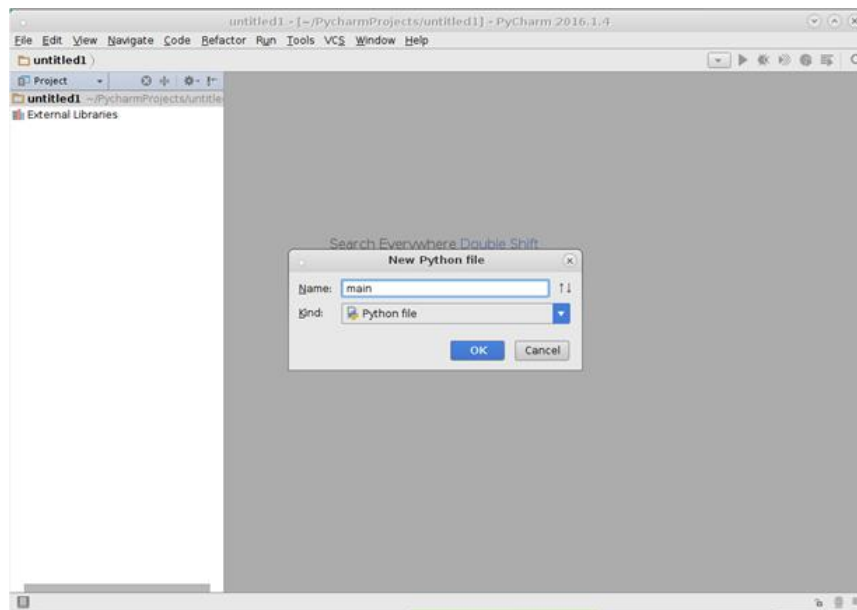


Рис. 68. Задаём название файла из букв латинского алфавита и цифр

Проект создан, можно писать код программы. Также в проект можно добавить уже существующие файлы с кодом. Можно просто перенести необходимый файл в каталог проекта. После чего он сразу отобразится в файлах проекта.

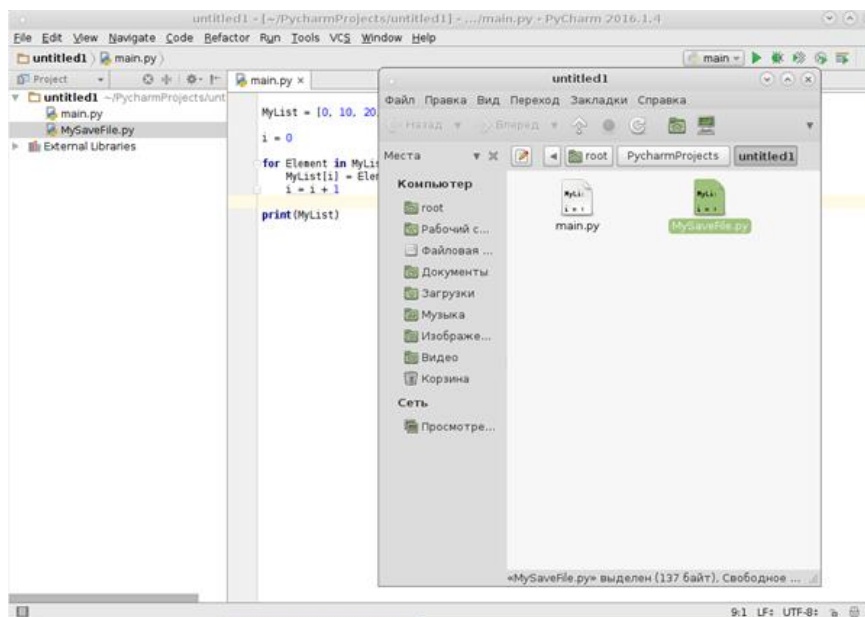


Рис. 69. Добавление файлов в проект

Интерфейс и выполнение проекта

Окно программы можно разделить на 4 области (Рис. 70):

1. в красной отображаются файлы проекта.
2. В зеленой происходит непосредственная работа с файлами, в том числе пишется код.
3. Синяя — служит для быстрого доступа к функциям построения, запуска и отладки проекта. Дополнительно присутствует функция поиска по коду.
4. Фиолетовая область — консоль вывода.

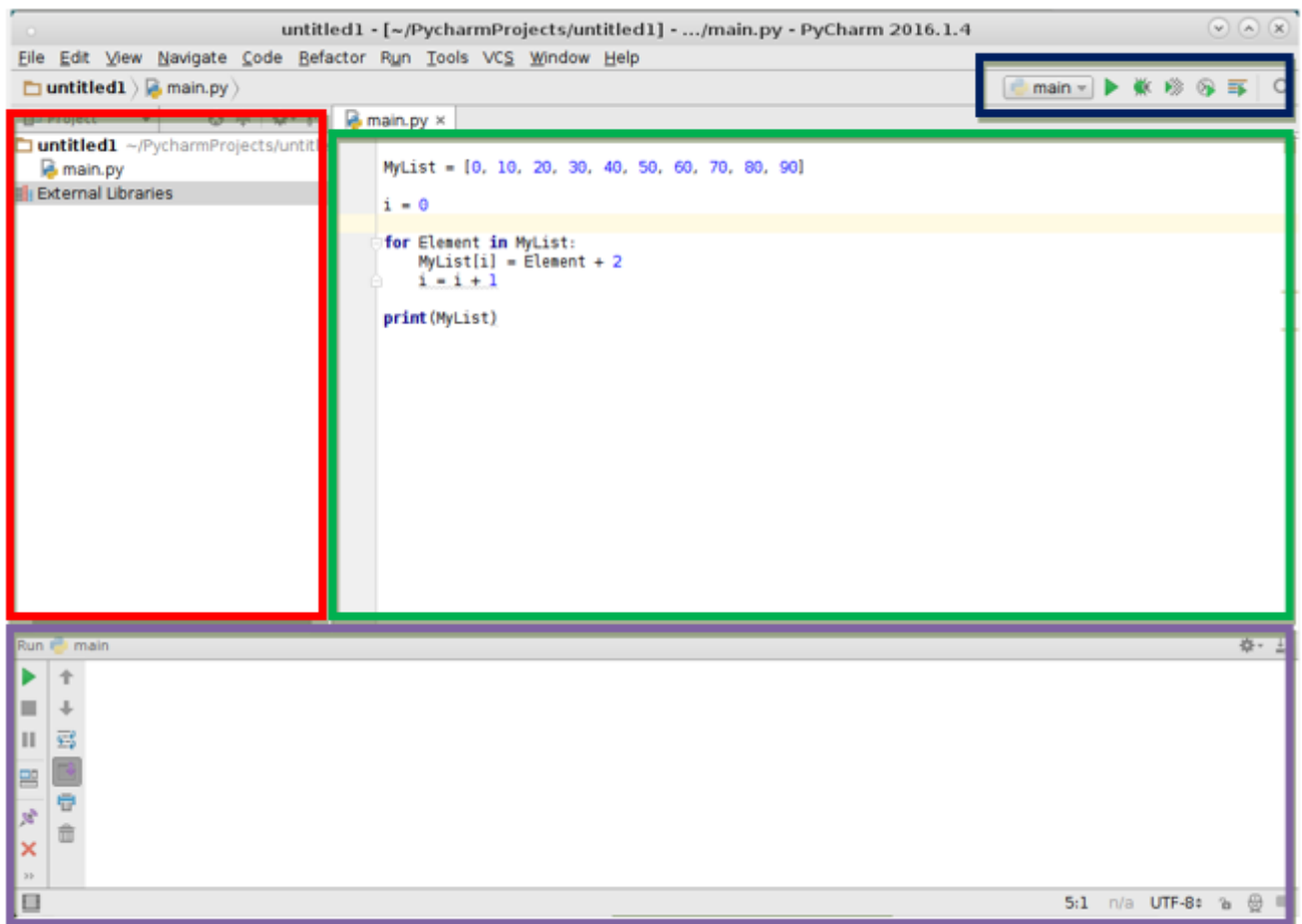


Рис. 71. Рабочие области PyCharm. Здесь приведена программа, которая увеличивает значения каждого элемента массива на 2.

Чтобы выполнить проект, в главном меню сделать Run→Run или использовать сочетание клавиш *Alt+Shift+F10* (Рис. 72-73).

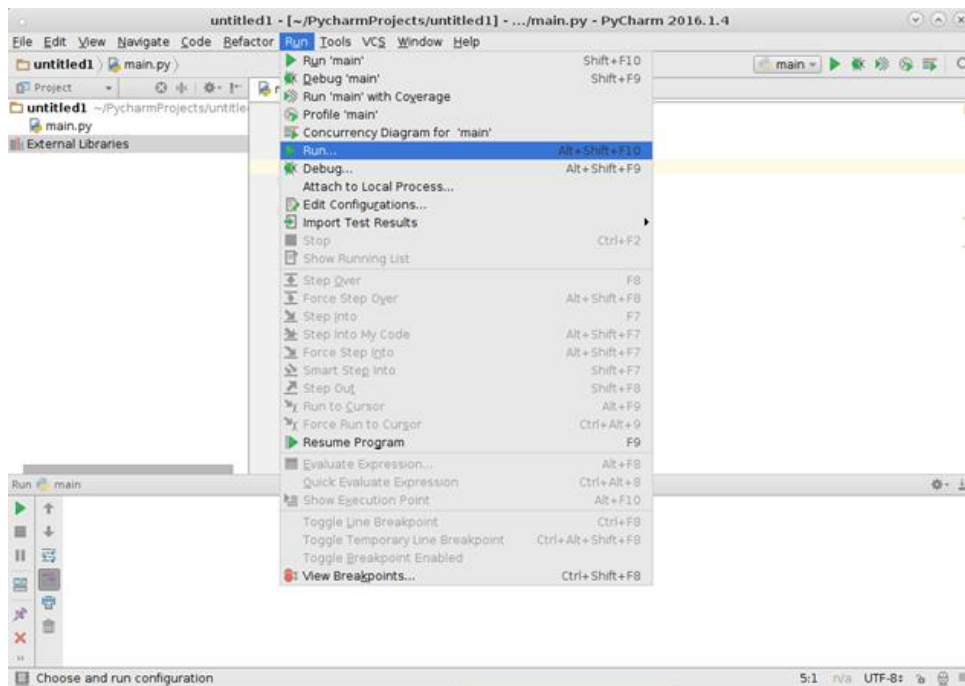


Рис. 74. Запускаем свою программу

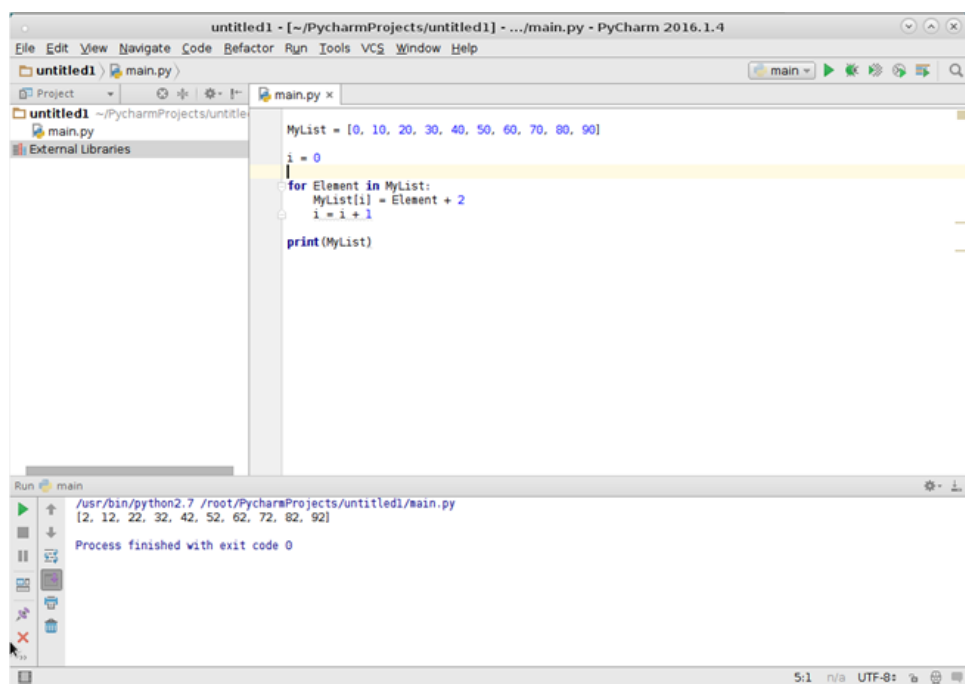


Рис. 75. Результат выполнения программы на вкладке вывода

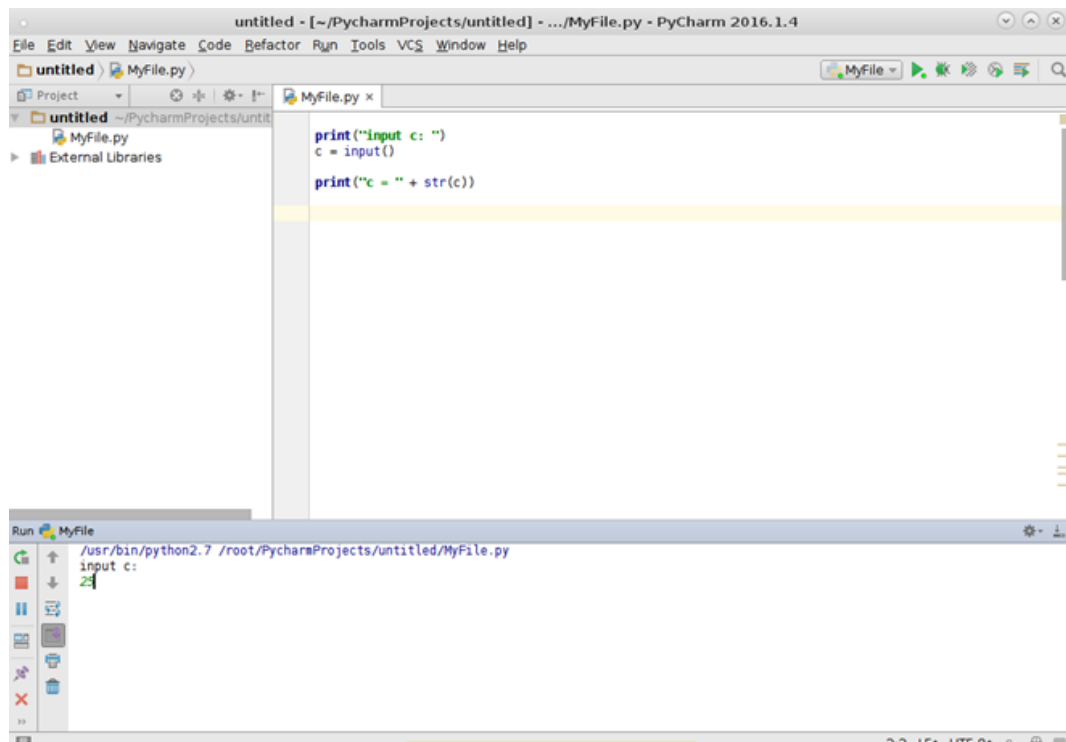


Рис. 76. Используем консоль для ввода данных

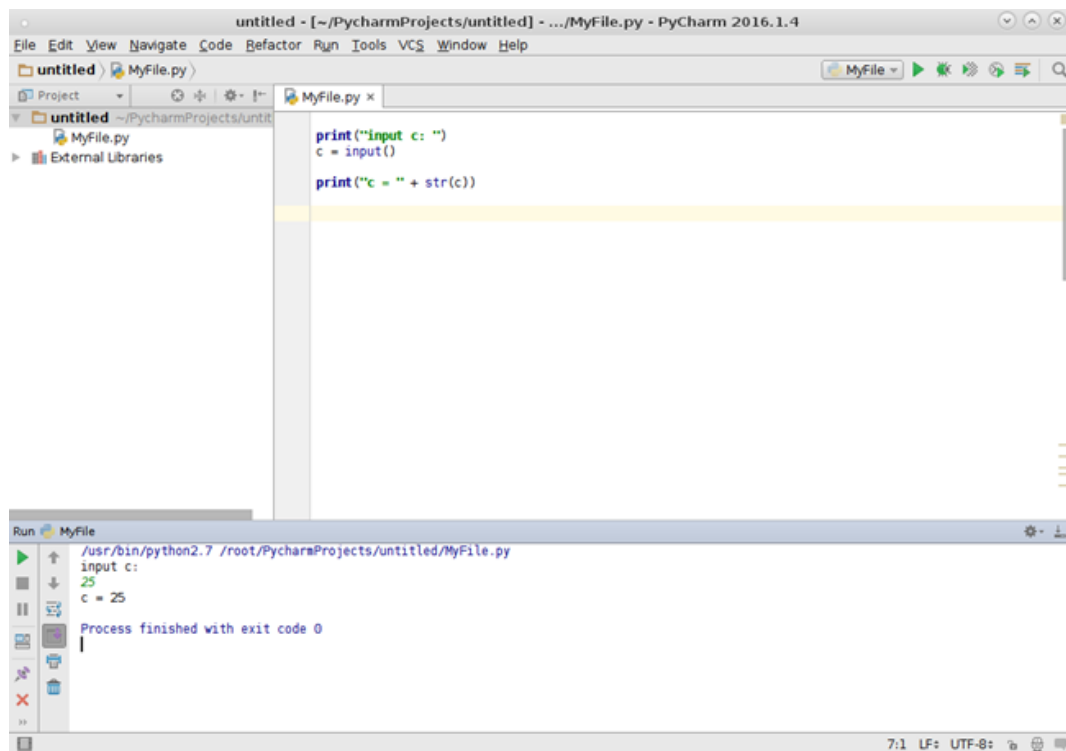


Рис. 77. Результат вычислений

Отладка проекта

Для отладки своего приложения можно пользоваться «breakpoint» (точки остановки программы), см. Рис. 78. Для того чтобы поставить «breakpoint» нужно нажать левую кнопку мыши слева от строки кода, на которой нужно остановить программу.

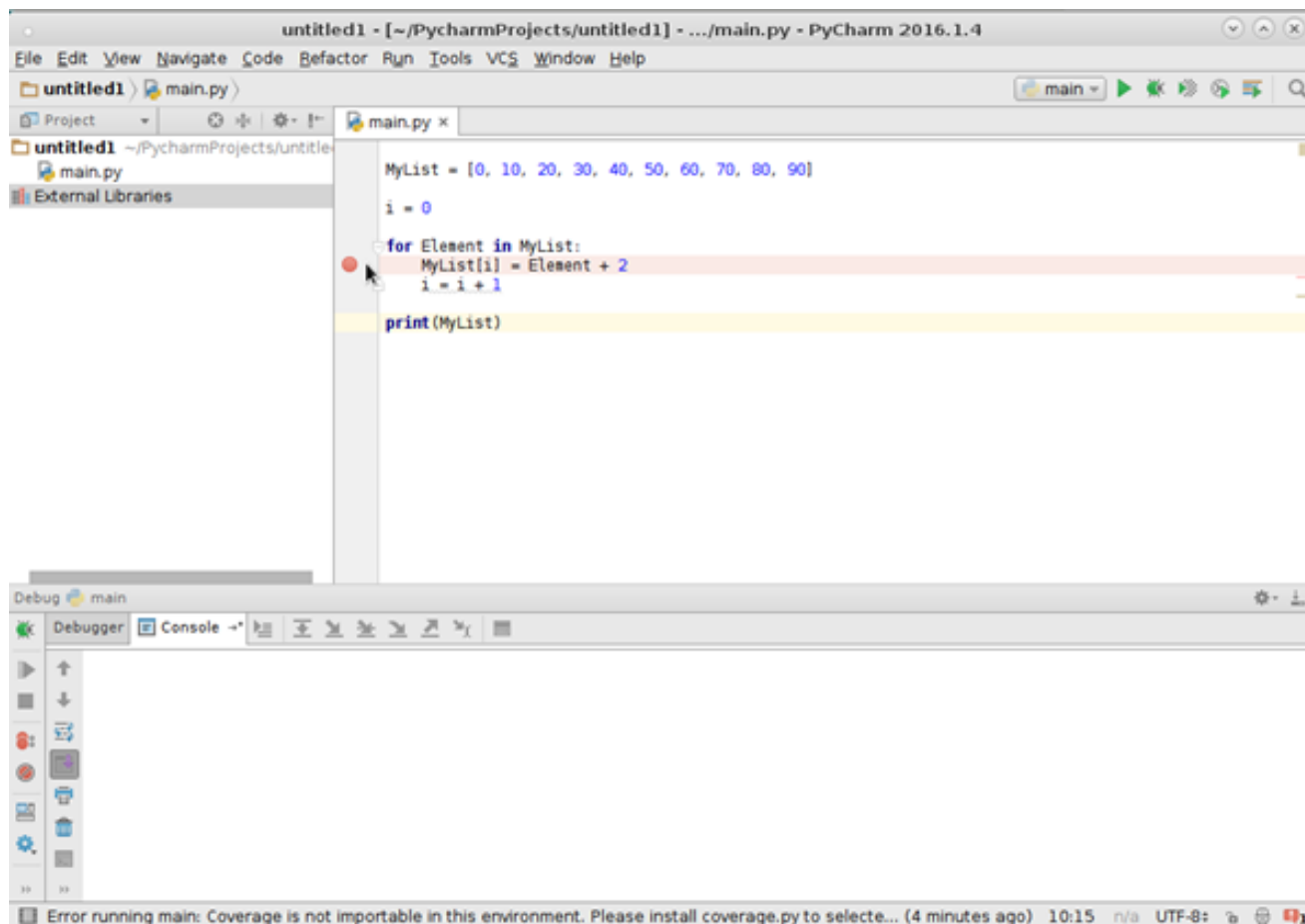


Рис. 79. Вставка точки остановки программы, *breakpoint*

После чего выбрать пункт главного меню *Run*→ *Debug* (Рис. 80).

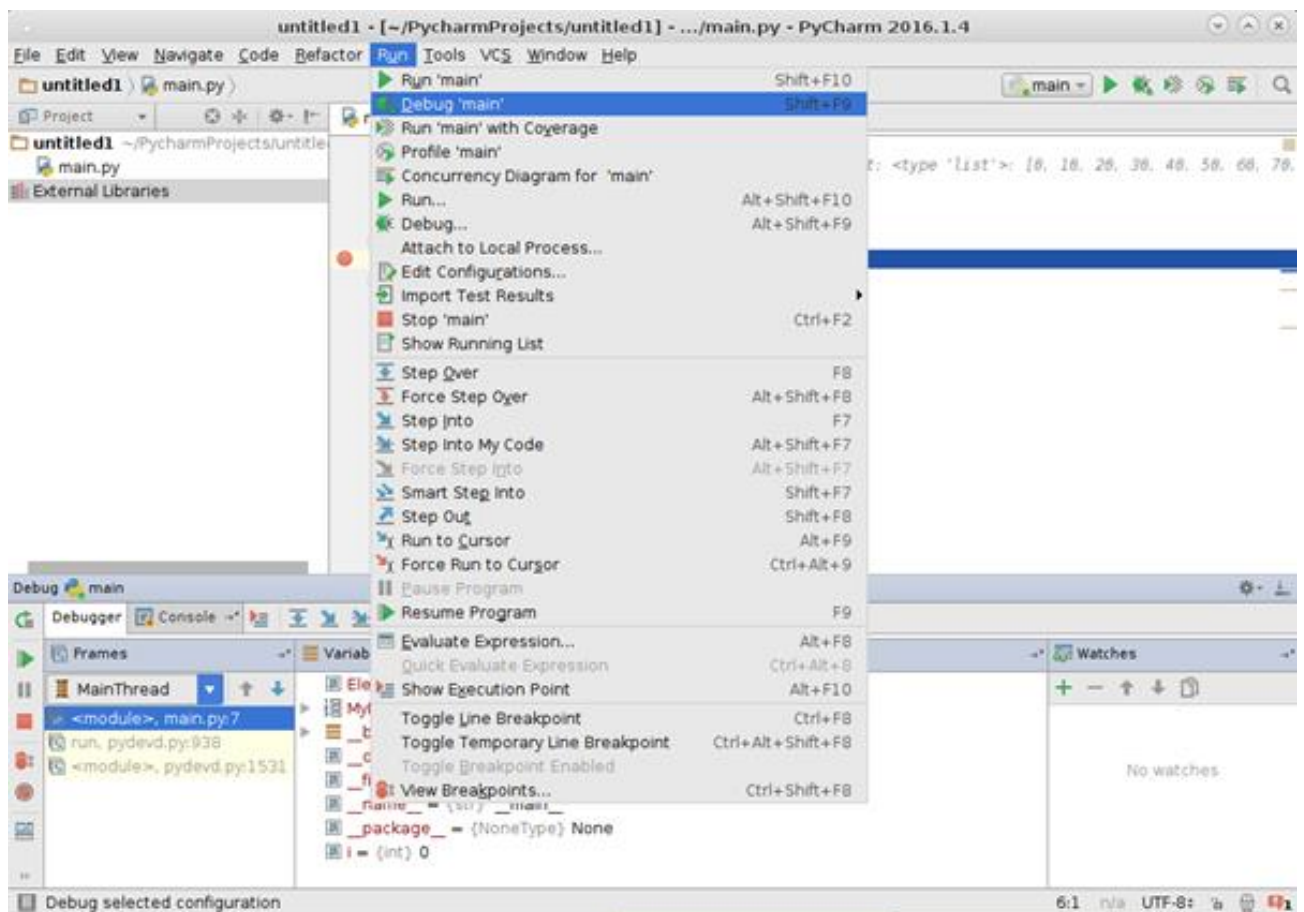


Рис. 81. Отладка — точка остановки программы

Программа остановит свое выполнение на указанном месте, после чего на вкладке *Debugger* в области *Variables* (зеленая область на рисунке ниже) можно отслеживать значения всех переменных. Также значения переменных можно увидеть непосредственно в окне с кодом программы (Рис. 82).

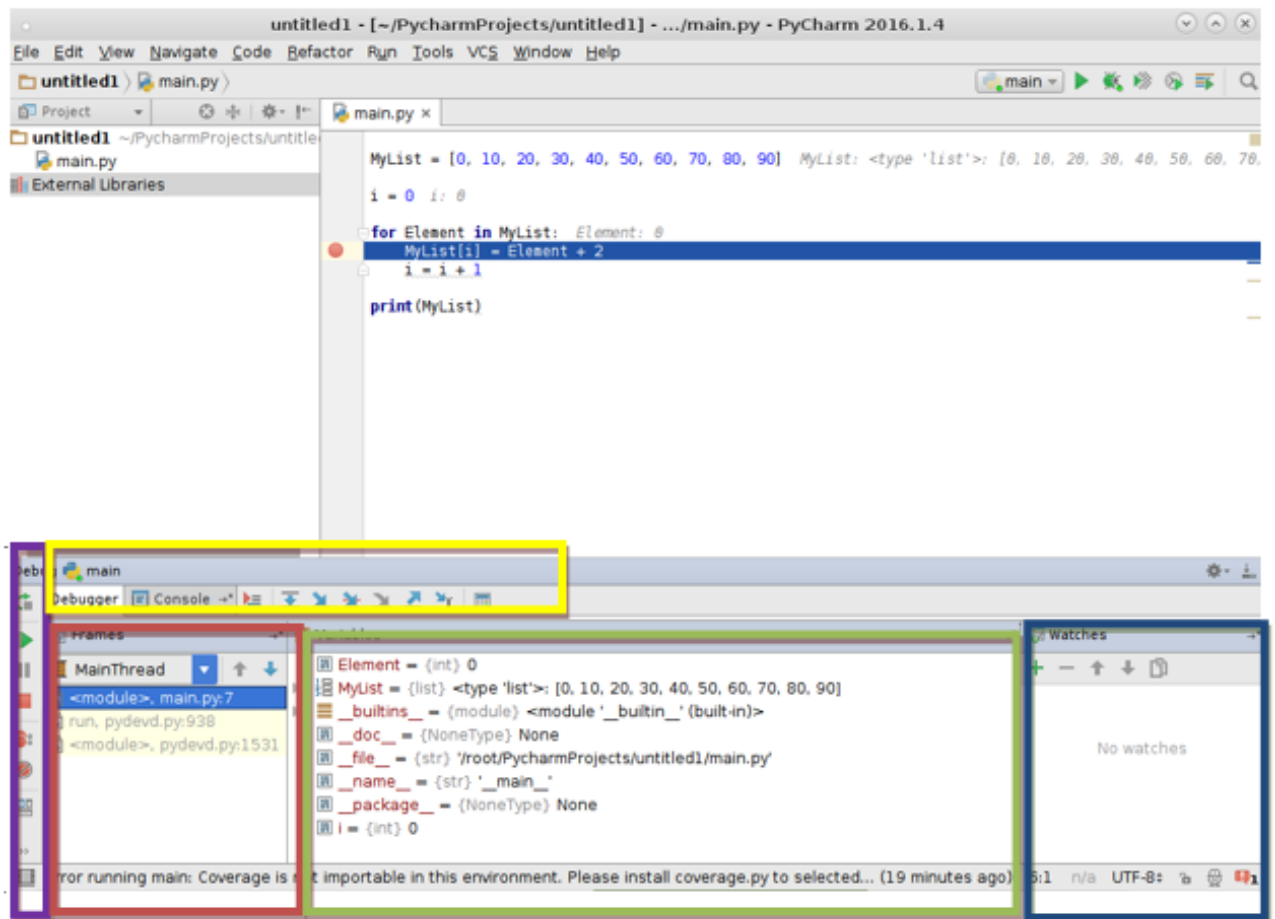


Рис. 83. Контроль значений переменных в точке останова программы

Фиолетовая область служит для быстрого доступа к необходимым функциям при отладке (от верхней к нижней кнопки):

- перезапуск проекта,
- выполнить программу до следующей точки прерывания,
- пауза (не активна в данный момент),
- принудительное завершение программы,
- просмотр выставленных «брейкпоинтов»,
- игнорировать «брейкпоинты» (программа будет выполнена до конца).

Желтая область служит для переключения между вкладками отладчика и консолью вывода, также в ней находятся кнопки для отладки:

- шаг с заходом,

- шаг с обходом,
- шаг с выходом.

Панель Frames (красная область) позволяет получить доступ к списку потоков вашего приложения. Для каждого потока, вы можете просматривать структуру стека, изучить кадры, перемещаться между кадрами, и автоматически переходить к исходному коду в редакторе.

Если нужно отследить значение определенной переменной, можно воспользоваться окном Watches. Чтобы добавить интересующую Вас переменную необходимо нажать на зеленый плюс и вписать название переменной из кода программы.

Аналогично можно добавить любую другую переменную. Например, чтобы добавить массив, так же нужно указать его имя, после чего во вкладке watches появятся значения элементов массива. Далее скриншоты процесса отладки, см. Рис. 84-85:

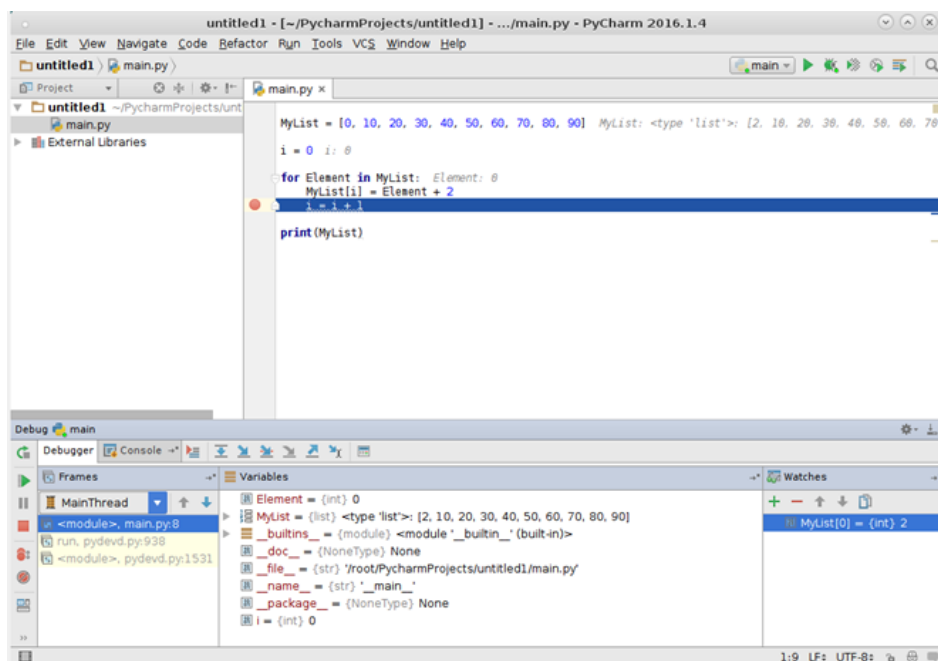


Рис. 86. Контроль значения переменной

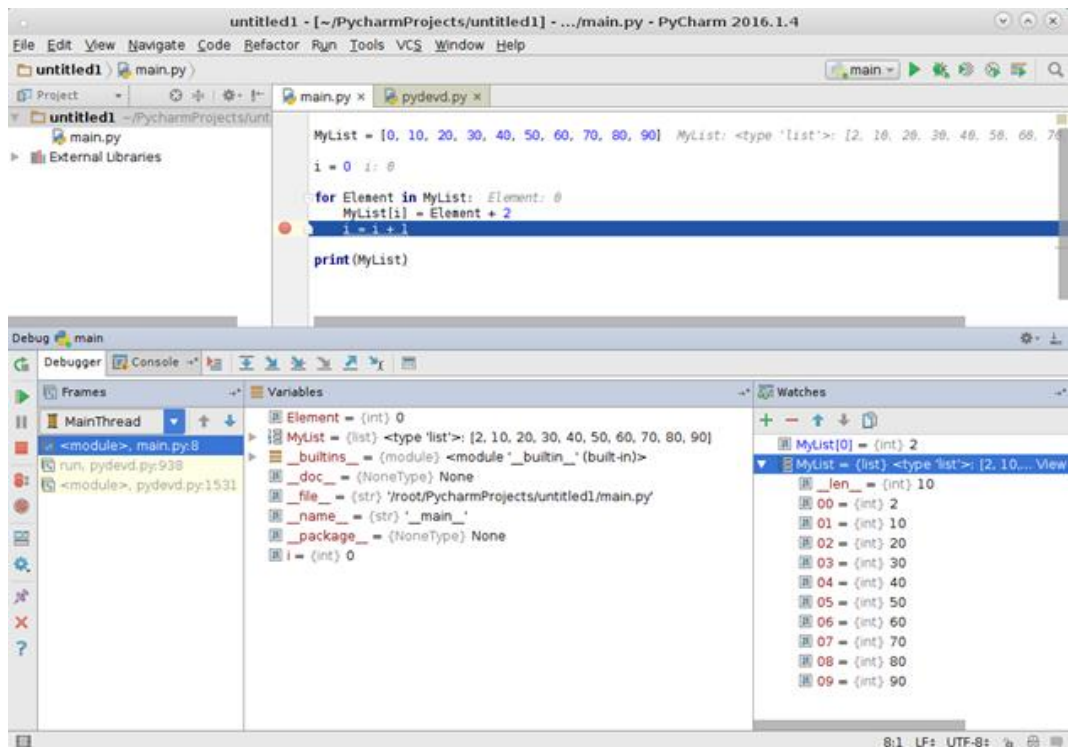


Рис. 87. Контроль значений элементов всего массива

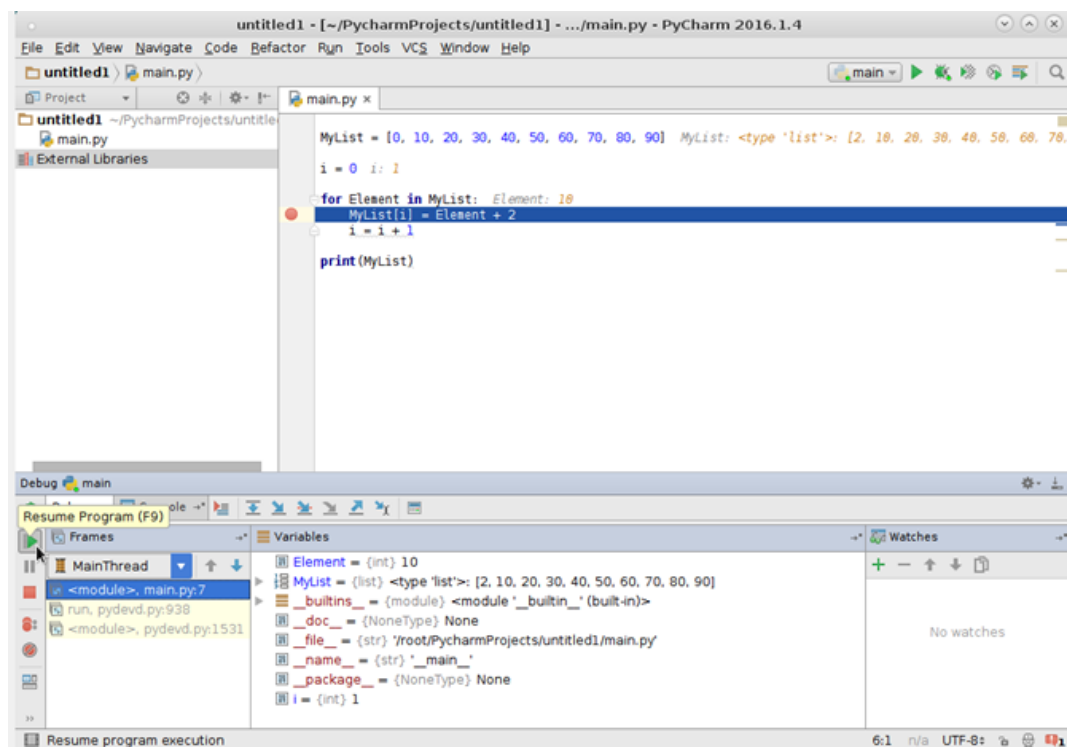


Рис. 88. Следующий шаг программы — кнопка Resume program или клавиша F9

Таким образом, можно контролировать ход выполнения программы. Если значение переменной больше отслеживать не требуется, то можно удалить ее из списка переменных. Для удаления используется значок красного минуса.

Разминка. 70 задач для обретения уверенности

Во время разминки необходимо создать следующие проекты в соответствии с разделами задач:

1. *linear_algorithms* — линейные алгоритмы (6 задач);
2. *branching_selection_operators* — ветвления и операторы выбора (23 задачи);
3. *cyclic_algorithms* — циклические алгоритмы (37 задач);
4. *associative_array* — ассоциативные массивы (4 задачи).

В один файл проекта можно записывать один или несколько алгоритмов. Если в файл записано решение одной задачи, то имя его должно соответствовать номеру задачи, например, *56.py*. Если в файле несколько задач, то в имени записывается диапазон номеров, например, *35_38.py*, или список номеров всех задач, например, *35_36_38_42.py*. Таким образом, последний файл в проекте *associative_array* будет называться *70.py*.

В каждом файле проектов обязательно должен быть заголовок, например,

#имя проекта: cyclic_algorithms

#номер версии: 1.0

#имя файла: 33_39.py

#автор и его учебная группа: Е. Волков, ЭУ-142

#дата создания: 20.03.2019

дата последней модификации: 25.03.2019

#связанные файлы: нет

описание: циклические алгоритмы

Для обретения уверенности необходимо решить **все 70 задач самостоятельно**. В противном случае толку не будет — получить зачёт и обмануть преподавателя можно, обрести уверенность никак нельзя. Кстати, во время разминки рекомендую активно пользоваться шпаргалками по Python, см. Приложения 1 и 2.

Приступаем:

Линейные алгоритмы

1. Преобразовать системную дату в «русский» формат представления дат, день/месяц/год (например, 17/05/2009).

Указание. Системная дата имеет вид 2009-06-15. Нужно преобразовать это значение в строку, строку разделить на компоненты (символ→разделитель→дефис), потом из этих компонентов сконструировать нужную строку.

2. Даны действительные числа — A, B, C. Найти максимальное и минимальное из этих чисел.
3. Известны длины трёх сторон треугольника. Вычислить периметр треугольника и площадь по формуле Герона (указание: использовать модуль math и функцию sqrt()).
4. Задан вес в граммах. Определить вес в тоннах и килограммах.
5. Известен объем информации в байтах. Перевести в килобайты, мегабайты.
6. Определить значение функции $Z = 1 / (X \times Y)$ при $(X \vee Y) \neq 0$.

Ветвления и оператор выбора

7. Дано натуральное число. Определить, будет ли это число: чётным, кратным 4.
8. Дано натуральное число. Определить, будет ли это число: нечётным, кратным 5.

9. Дано натуральное число. Определить, будет ли это число: нечётным, кратным 7.
10. Дано натуральное число. Определить, будет ли это число: чётным, кратным 10.
11. Имеется коробка со сторонами: $A \times B \times C$. Определить, пройдёт ли она в дверь с размерами $M \times K$.
12. Дано вещественное число. Определить, какое это число: положительное, отрицательное, ноль.
13. Можно ли из бревна, имеющего диаметр поперечного сечения D , выпилить квадратный брус шириной A ?
14. Можно ли в квадратном зале площадью S поместить круглую сцену радиусом R так, чтобы от стены до сцены был проход не менее K ?
15. Дан номер места в плацкартном вагоне. Определить, какое это место: верхнее или нижнее, в купе или боковое.
16. Известна денежная сумма. Разменять её купюрами 500, 100, 10 и монетой 2 руб., если это возможно.
17. Имеются две ёмкости: кубическая с ребром A , цилиндрическая с высотой H и радиусом основания R . Определить, поместится ли жидкость объёма M в первую ёмкость, во вторую, в обе.
18. Имеются две ёмкости: кубическая с ребром A , цилиндрическая с высотой H и радиусом основания R . Определить, можно ли заполнить жидкостью объёма M первую ёмкость, вторую, обе.
19. Даны вещественные числа: X , Y , Z . Определить, существует ли треугольник с такими длинами сторон и, если существует, будет ли он прямоугольным.
20. Дано число X . Определить, принадлежит ли это число заданному промежутку $[a, b]$.
21. Определить значение функции $Z = 1/(XY)$ при произвольных X и Y .

22. Даны вещественные числа: A, B, C . Определить, выполняются ли неравенства $A < B < C$ или $A > B > C$ и какое именно неравенство выполняется.
23. Даны вещественные числа X и Y . Вычислить Z . $Z = \sqrt[3]{(X \times Y)}$ при $X > Y$, $Z = \ln(X + Y)$ в противном случае.
24. Даны вещественные положительные числа a, b, c, d . Выясните, может ли прямоугольник со сторонами a, b уместиться внутри прямоугольника со сторонами c, d так, чтобы каждая сторона внутреннего прямоугольника была параллельна или перпендикулярна стороне внешнего прямоугольника.
25. Дано вещественное число A . Вычислить $f(A)$, если $f(x) = x^2 + 4x + 5$, при $x \leq 2$; в противном случае $f(x) = 1/(x^2 + 4x + 5)$.
26. Дано вещественное число A . Вычислить $f(A)$, если $f(x) = 0$, при $x \leq 0$; $f(x) = x$ при $0 < x < 1$, в противном случае $f(x) = x^4$.
27. Дано вещественное число A . Вычислить $f(A)$, если $f(x) = 0$ при $x \leq 0$; $f(x) = x^2 - x$ при $0 < x < 1$, в противном случае $f(x) = x^2 - \sin(\pi x^2)$.
28. Составить алгоритм и программу для реализации логических операций «И» и «ИЛИ» для двух переменных.
29. Известен ГОД. Определить, будет ли этот год високосным, и к какому веку этот год относится

Указание. При вычислении корней и логарифмов используйте функции `sqrt()` и `log()` модуля `math`. В этом же модуле определена константа `pi` (`math.pi`).

Циклические алгоритмы. Обработка последовательностей и одномерных массивов (списков)

30. Составьте блок-схему поиска максимального элемента в одномерном массиве.
31. Нарисуйте полную блок-схему алгоритма сортировки массива «методом пузырька».

32. Дан список числовых значений, насчитывающий N элементов. Поменять местами элементы, стоящие на чётных и нечётных местах:
 $A[1] \leftrightarrow A[2]; A[3] \leftrightarrow A[4] \dots$
33. Дан список числовых значений, насчитывающий N элементов. Выполнить перемещение элементов массива по кругу вправо, т. е.
 $A[1] \rightarrow A[2]; A[2] \rightarrow A[3]; \dots A[n] \rightarrow A[1]$.
34. Дан список числовых значений, насчитывающий N элементов. Поменять местами первую и вторую половины массива.
35. Дан список числовых значений, насчитывающий N элементов. Поменять местами группу из M элементов, начинающихся с позиции K с группой из M элементов, начинающихся с позиции P .
36. Дан список числовых значений, насчитывающий N элементов. Вставить группу из M новых элементов, начиная с позиции K .
37. Дан список числовых значений, насчитывающий N элементов. Сумму элементов массива и количество положительных элементов поставить на первое и второе место.
38. Дан список числовых значений, насчитывающий N элементов. Исключить из него M элементов, начиная с позиции K .
39. Дан список числовых значений, насчитывающий N элементов. Исключить все нулевые элементы.
40. Дан список числовых значений, насчитывающий N элементов. После каждого отрицательного элемента вставить новый элемент, равный квадрату этого отрицательного элемента.
41. Дан список числовых значений, насчитывающий N элементов. Определить, образуют ли элементы массива, расположенные перед первым отрицательным элементом, возрастающую последовательность.

42. Дан список числовых значений, насчитывающий N элементов. Определить, образуют ли элементы массива, расположенные перед первым отрицательным элементом, убывающую последовательность.
43. Дан список числовых значений, насчитывающий N элементов. Из элементов исходного массива построить два новых. В первый должны входить только элементы с положительными значениями, а во второй — только элементы с отрицательными значениями.
44. Дан список числовых значений, насчитывающий N элементов. Добавить столько элементов, чтобы элементов с положительными и отрицательными значениями стало бы поровну.
45. Дан список числовых значений, насчитывающий N элементов. Добавить к элементам массива такой новый элемент, чтобы сумма элементов с положительными значениями стала бы равна модулю суммы элементов с отрицательными значениями.
46. Дан список числовых значений, насчитывающий N элементов. Дано положительное число T . Разделить это число между положительными элементами массива пропорционально значениям этих элементов и добавить полученные доли к соответствующим элементам.
47. Дан список числовых значений, насчитывающий N элементов. Исключить из массива элементы, принадлежащие промежутку $[B; C]$.
48. Дан список числовых значений, насчитывающий N элементов. Вместо каждого элемента с нулевым значением поставить сумму двух предыдущих элементов массива.
49. Дан список числовых значений, насчитывающий N элементов. Определить, имеются ли в массиве два подряд идущих нуля.
50. Дан список числовых значений, насчитывающий N элементов. Подсчитать количество чисел, делящихся на 3 нацело, и среднее арифметическое чисел с чётными значениями. Поставить полученные

величины на первое и последнее места в массиве (увеличив массив на 2 элемента).

51. Заданы М строк символов, которые вводятся с клавиатуры. Найти количество символов в самой длинной строке. Выровнять строки по самой длинной строке, поставив перед каждой строкой соответствующее количество звёздочек.
52. Заданы М строк символов, которые вводятся с клавиатуры. Из заданных строк, каждая из которых представляет одно слово, составить одну длинную строку, разделяя слова пробелами.
53. Заданы М строк слов, которые вводятся с клавиатуры. Подсчитать количество гласных букв в каждой из заданных строк.
54. Заданы М строк слов, которые вводятся с клавиатуры (в каждой строке — одно слово). Вводится слог (последовательность букв). Подсчитать количество таких слогов в каждой строке.
55. Заданы М строк слов, которые вводятся с клавиатуры (в каждой строке — одно слово). Вводится слог (последовательность букв). Удалить данный слог из каждой строки.
56. Заданы М строк символов, которые вводятся с клавиатуры. Напечатать все центральные буквы строк нечетной длины.
57. Заданы М строк символов, которые вводятся с клавиатуры. Каждая строка содержит слово. Записать каждое слово в разрядку (вставить по пробелу между буквами).
58. Задана строка символов, в которой встречается символ «.». Поставить после каждого такого символа системное время ПК.
59. Заданы М строк, которые вводятся с клавиатуры. Подсчитать количество пробелов в каждой из строк.
60. Заданы М строк символов, которые вводятся с клавиатуры. Каждая строка представляет собой последовательность символов,

включающих в себя вопросительные знаки. Заменить в каждой строке все имеющиеся вопросительные знаки звёздочками.

61. Последовательно вводятся числа. Определить сумму чисел с нечётными номерами и произведение чисел с чётными номерами (по порядку ввода). Подсчитать количество слагаемых и количество сомножителей. При вводе числа 55555 закончить работу.
62. Определить сумму вводимых положительных чисел. Причём числа с нечётными номерами (по порядку ввода) суммировать с обратным знаком, а числа с чётными номерами перед суммированием возводить в квадрат. Подсчитать количество слагаемых. При вводе первого отрицательного числа закончить работу.
63. Дано число P и число N . Определить сумму чисел меньше P , произведение чисел больше N и количество чисел в диапазоне значений P и N . При вводе числа равного P или N , закончить работу.
64. Суммировать вводимые числа, среди которых нет нулевых. При вводе нуля обеспечить вывод текущего значения суммы. При вводе числа 99999 закончить работу.
65. Вводятся положительные числа. Определить сумму чисел, делящихся на положительное число V нацело. При вводе отрицательного числа закончить работу.
66. Для вводимых чисел определить процент положительных и отрицательных чисел. При вводе числа -65432 закончить работу.

Работа с ассоциативными массивами (таблицами данных)

67. Используя данные таблицы отсортировать блюда по возрастанию цены. Вывести отсортированный вариант списка блюд.

Блюдо	Цена
-------	------

Борщ	35
Котлета	40
Каша	20
Чай	3

68. Имеется список учеников и результаты трёх тестов (баллы от 0 до 100). Определить средний балл каждого ученика по трём тестам, вывести список учеников по убыванию среднего балла.
69. Известны данные о количестве мальчиков и девочек в нескольких классах. Отсортировать названия классов по возрастанию процента мальчиков, определить количество классов, в которых мальчиков больше, чем девочек, и вывести названия этих классов отдельно.
70. Решить задачу, связанную с оценкой экономической деятельности группы предприятий на основе известных данных:
- название предприятий;
 - плановый объем розничного товарооборота;
 - фактический объем розничного товарооборота.

Требуется определить:

- а) процент выполнения плана каждым предприятием;
- б) количество предприятий, недовыполнивших план на 10% и более;
- с) наименьший плановый товарооборот;
- д) упорядочить предприятия по убыванию планового товара.

Курсовая работа — часть 1. Экспериментальная оценка эффективности алгоритмов внутренней сортировки

“... Сортировка к тому же, еще и сама достаточно хороший пример задачи, которую можно решать с помощью многих различных алгоритмов. Каждый из них имеет и свои достоинства, и свои недостатки, и выбирать алгоритмы нужно исходя из конкретной постановки задачи.

В общем, сортировку следует понимать, как процесс перегруппировки заданного множества объектов в некотором определенном порядке. Цель сортировки — облегчить последующий поиск элементов в таком отсортированном множестве. Это почти универсальная, фундаментальная деятельность. Мы встречаемся с отсортированными объектами в телефонных книгах, в списках подходящих налогов, в оглавлениях книг, в библиотеках, в словарях, на складах — почти везде, где нужно искать хранимые объекты. Даже малышей учат держать свои вещи «в порядке», и они уже сталкиваются с некоторыми видами сортировок задолго до того, как познакомятся с азами арифметики”³.

Итак, задача сортировки ставится следующим образом: имеется последовательность однотипных записей — строк, разделенных на поля, в которые можно записывать данные различных типов. Одно из полей записей выбрано в качестве ключевого, далее будем называть его ключом сортировки. Необходимо чтобы тип данных ключа допускал операции сравнения ("равно", "больше", "меньше", "не больше" и "не меньше"). Как правило, ключом сортировки являются данные целого типа. Задачей сортировки является преобразование исходной последовательности в последовательность, содержащую те же записи, но в порядке возрастания или убывания значений ключа. Метод сортировки называется *устойчивым*, если при его применении не изменяется относительное положение записей с равными значениями ключа.

Различают сортировку массивов записей, целиком расположенных в основной памяти — внутреннюю сортировку, и сортировку файлов, хранящихся во внешней памяти и не помещающихся полностью в основной памяти — внешнюю сортировку. Для внутренней и внешней сортировки требуются различные методы. В нашей лабораторной работе будем изучать наиболее известные, простые и понятные методы внутренней сортировки, используя средства языка программирования Python.

Требованием, предъявляемое к внутренней сортировке является то, что методы не должны требовать дополнительной памяти: все перестановки с целью

³ Вирт, Н. Алгоритмы и структуры данных / Н. Вирт. — М: Изд-во «Мир», 1989. — 360 с.;

упорядочения элементов массива должны производиться в пределах того же массива. Мерой эффективности алгоритма внутренней сортировки являются число требуемых операций сравнения значений ключа (от английского Compare — C) и число перестановок элементов (от английского Move — M).

Заметим, что поскольку сортировка основана только на значениях ключа и никак не затрагивает оставшиеся поля записей, можно говорить о сортировке массивов ключей. Таким образом, задачу сортировки можно сформулировать следующим образом: задан одномерный целочисленный массив **arr[N]** размером $N = \text{dim}$, необходимо расположить элементы этого массива в порядке возрастания или убывания значений.

Сортировка включением

Одним из наиболее простых и естественных методов внутренней сортировки является сортировка простыми включениями. Идея алгоритма очень проста. Пусть имеется массив ключей **arr[0], arr[1], ..., arr[N-1]**. Для каждого элемента массива, начиная со второго, производится сравнение с элементами с меньшим индексом. Элемент **arr[i]** последовательно сравнивается с элементами **arr[j]**, где $j \in [i-1; 0]$, т.е. изменяется от $i-1$ до 0 . До тех пор, пока для очередного элемента **arr[j]** выполняется соотношение **arr[j] > arr[i]**, **arr[i]** и **arr[j]** меняются местами. Если удастся встретить такой элемент **arr[j]**, что **arr[j] ≤ arr[i]**, или если достигнута нижняя граница массива, производится переход к обработке элемента **arr[i+1]**. Так продолжается до тех пор, пока не будет достигнута верхняя граница массива.

Легко видеть, что в лучшем случае, когда массив уже упорядочен для выполнения алгоритма с массивом из N элементов потребуется $N-1$ сравнение и 0 пересылок. В худшем случае, когда массив упорядочен в обратном порядке потребуется $N(N-1)/2$ сравнений и столько же пересылок. Таким образом, можно оценивать сложность метода простых включений как $O(N^2)$.

Можно сократить число сравнений, применяемых в методе простых включений, если воспользоваться тем, что при обработке элемента **arr[i]** массива элементы **arr[0], arr[1], ..., arr[i-1]** уже упорядочены, и воспользоваться для поиска элемента, с которым должна быть произведена перестановка, методом двоичного деления. В этом случае оценка числа требуемых сравнений становится $O(N \log N)$. Заметим, что поскольку при выполнении перестановки требуется сдвигка на один элемент нескольких элементов, то оценка числа пересылок остается $O(N^2)$. Алгоритм сортировки включением, оформленный в виде функции приведен в файле **insert.py** и в приложении 1.

Обменная сортировка

Простая обменная сортировка, называемая «методом пузырька», для массива $arr[0], arr[2], \dots, arr[N-1]$ работает следующим образом. Начиная с конца массива сравниваются два соседних элемента $arr[N-1]$ и $arr[N-2]$. Если выполняется условие $arr[N-2] > arr[N-1]$, то они меняются местами. Процесс продолжается для $arr[N-2]$ и $arr[N-3]$ и т.д., пока не будет произведено сравнение $arr[1]$ и $arr[0]$. Понятно, что после этого на месте $arr[0]$ окажется элемент с наименьшим значением. На втором шаге процесс повторяется, но последними сравниваются $arr[2]$ и $arr[1]$. И так далее. На последнем шаге будут сравниваться только текущие значения $arr[N-1]$ и $arr[N-2]$. Понятна аналогия с пузырьком, поскольку наименьшие элементы, самые «легкие», постепенно «всплывают» к верхней границе массива.

Для метода обменной сортировки требуется число сравнений $N(N-1)/2$, минимальное число пересылок 0, а среднее и максимальное число пересылок — $O(N^2)$.

Метод пузырька допускает три простых усовершенствования. Во-первых, если на некотором шаге не было произведено ни одного обмена, то выполнение алгоритма можно прекращать. Во-вторых, можно запоминать наименьшее значение индекса массива, для которого на текущем шаге выполнялись перестановки. Очевидно, что верхняя часть массива до элемента с этим индексом уже отсортирована, и на следующем шаге можно прекращать сравнения значений соседних элементов при достижении такого значения индекса. В-третьих, метод пузырька работает неравноправно для «легких» и «тяжелых» значений. Легкое значение попадает на нужное место за один шаг, а тяжелое на каждом шаге опускается по направлению к нужному месту на одну позицию.

Сортировка выбором

При сортировке массива $arr[0], arr[2], \dots, arr[N-1]$ методом простого выбора среди всех элементов находится элемент с наименьшим значением $arr[i]$, и $arr[0]$ и $arr[i]$ обмениваются значениями. Затем этот процесс повторяется для получаемого подмассива $arr[1], arr[2], \dots, arr[N-1], \dots arr[j], arr[j+1], \dots, arr[N-1]$ до тех пор, пока мы не дойдем до подмассива $arr[N-1]$, содержащего к этому моменту наибольшее значение.

Для метода сортировки простым выбором оценка требуемого числа сравнений — $N(N-1)/2$. Порядок требуемого числа пересылок, которые требуются для выбора минимального элемента, в худшем случае составляет $O(N^2)$. Однако порядок среднего числа пересылок есть $O(N \lg N)$, что в ряде случаев делает этот метод предпочтительным.

Сравнение методов внутренней сортировки

Для рассмотренных простых методов сортировки существуют точные формулы, вычисление которых дает минимальное, максимальное и среднее число сравнений ключей C и пересылок элементов массива M . Таблица 2 содержит данные, приводимые в книге Никлааса Вирта «Алгоритмы + данные = программы»⁴.

Таблица 3. Теоретические характеристики эффективности методов сортировки

		Min	Avg	Max
Прямое включение	C	$N-1$	$(N^2 + N - 2)/4$	$(N^2 - N)/2 - 1$
	M	$2(N-1)$	$(N^2 - 9N - 10)/4$	$(N^2 - 3N - 4)/2$
Прямой выбор	C	$(N^2 - N)/2$	$(N^2 - N)/2$	$(N^2 - N)/2$
	M	$3(N-1)$	$N(\ln(N) + 0,57)$	$N^2/4 + 3(N-1)$
Прямой обмен	C	$(N^2 - N)/2$	$(N^2 - N)/2$	$(N^2 - N)/2$
	M	0	$0,75(N^2 - N)$	$1,5(N^2 - N)$

⁴ Вирт, Н. Алгоритмы и структуры данных / Н. Вирт. – М: Изд-во «Мир», 1989. – 360 с.;

Что надо сделать в первой части курсовой

Вам необходимо:

- 1) заполнить экспериментальными данными сравнительную таблицу эффективности алгоритмов (Таблица 4);
- 2) рассчитать теоретические значения показателей эффективности для массива вашего размера;
- 3) сравнить экспериментально полученные и расчетные значения;
- 4) высказать свое мнение об эффективности алгоритмов.

Таблица 5. Экспериментальная сравнительная эффективность алгоритмов сортировки

Массив	Показатель/Метод	Insert	Select	Bubble
Упорядоченный список	C			
	M			
Обратно упорядоченный список	C			
	M			
Одномерный массив со случайными значениями	Avr(C)			
	Avr(M)			

По структуре таблицы сравнения эффективности легко видеть, что Вам необходимо проделать как минимум 9 опытов, в каждом из которых должны быть определены два числа *C* и *M*. Но в третьем случае, массива со случайными значениями, в отличие от двух первых, заранее нельзя предсказать показатели эффективности (смотри теорию выше). Естественно, что сравнительный анализ алгоритмов возможен только по средним значениям (в таблице Вы видите функцию *Avr(C)* — *Average*, что значит среднее значение *C*). Для этого необходимо провести серию опытов, желательно, 100–150 для каждого алгоритма, т.е. всего 300–450 опытов. Именно это «жуткое» количество экспериментов и определяет структуру построения программы для реализации плана выполнения задания.

Итак, каждый алгоритм сортировки должен быть оформлен в виде самостоятельной программной единицы — функции, которая может быть

использована многократно. Записать эти функции необходимо в различные файлы, например, `insert.py`, `select.py` и `bubble.py`. Пример одной из них, метод простого включения, Вы найдете в файле `insert.py` и в приложении 1 в конце этих указаний, а вот оставшиеся две Вам надо написать самостоятельно. Обратите внимание на оформление текста программы и комментарии к алгоритму. Комментарии не позволят Вам забыть эти алгоритмы через неделю после сдачи отчета о своей работе. Кроме того, в отдельный файл (например, `sort-methods.py`) необходимо записать основную функцию), в которой реализован план части 1. Пример такой функции в файле `sort-methods.py` и в приложении 2 в конце пособия. Код несколько прямолинеен и, конечно, его можно оптимизировать, но за то он понятен, что сейчас самое главное.

Прочитаем внимательно комментарии, отбросив исходный код. Ниже приведен листинг программы без операторов языка Python — только комментарии.

```
# Попробуем открыть файл для сохранения результата
# 1) АНАЛИЗ УПОРЯДОЧЕННОГО МАССИВА
# Формируем исходный массив и выводим его на экран
# Последовательно обнуляем счетчики показателей эффективности
# SCount, MCount и вызываем функции select(...), insert(...) и bubble(...)
# Результаты выводим на экран и при успешном открытии файла записываем
# 2) АНАЛИЗ ОБРАТНО УПОРЯДОЧЕННОГО МАССИВА
# Формируем исходный массив и выводим его на экран
# Последовательно обнуляем счетчики показателей эффективности
# SCount, MCount и вызываем функции select(...), insert(...) и bubble(...)
# Результаты выводим на экран и при успешном открытии файла записываем.
# 3) АНАЛИЗ СЛУЧАЙНОГО МАССИВА
# Создаем цикл для нескольких реализаций случайного массива
# Формируем исходный массив
# Последовательно обнуляем счетчики показателей эффективности
# SCount, MCount и вызываем функции select(...), insert(...) и bubble(...)
# Результаты при успешном открытии файла записываем
```

*# Перед завершением программы закрываем файл и ждем нажатия клавиши.
Картинка остается на экране*

Легко видеть, что в основной программе решены нижеследующие задачи.

1. Распределение памяти для сохранения массив и показателей эффективности;
2. Автоматическое формирование массивов для всех 3-х экспериментальных случаев, а именно, упорядоченного, обратно упорядоченного и случайного;
3. Запись результатов работы программы на экран монитора и во внешний файл, если его имя задано в качестве параметра программы;
4. Последовательный вызов всех методов для случаев упорядоченного и обратно упорядоченного массивов;
5. Для случайного массива организован цикл, который позволяет получить требуемое количество реализаций случайной последовательности.

Последовательность выполнения работы

Оставшиеся три последних пункта первой части Вам необходимо отразить в отчете, а именно пункты:

- 2) рассчитать теоретические значения показателей эффективности для массива вашего размера. Проще всего это сделать с использованием MS Excell пакета MS Office, для различных значений размера массива;
- 3) сравнить экспериментально полученные и расчетные значения;
- 4) высказать свое мнение об эффективности алгоритмов и правильности теоретических формул.

Оформление отчёта по первой части

Надеюсь, фантазия Вас не подвела, и Вы успешно справились с заданием — пора подумать об отчёте и вспомнить преподавателя. Итак, в качестве отчетных материалов необходимо предоставить нижеследующее.

1. Красиво оформленные исходные тексты Ваших программ, выполненные в едином стиле. О стиле записи исходных текстов прочитайте в работе [2] библиографии. Не забывайте — в программах необходимы комментарии. Каждый файл должен обязательно содержать «шапку», в которой указываются:

- имя проекта,
- имя файла,
- номер версии,
- автор и его учебная группа,
- комментарии с кратким описанием модуля,
- список всех файлов для совместной сборки программы,
- даты создания и
- последней модификации кода.

Не забывай комментировать код. В первой части курсовой должно быть пять файлов: 3 функции, реализующие алгоритмы сортировки и общая программа для получения статистики при экспериментировании, плюс результаты эксперимента со случайными массивами;

2. Загружаемые программы, собранные без подключения пакетов и run-time библиотек. Всего таких файлов 4;

3. Отчет с обработкой результатов экспериментов в формате MS Office и выводами о сравнении теории с практикой;

4. По результатам работы проводится зачетный тест. Примерные вопросы теста приведены ниже в разделе вопросы для самоконтроля.

Вопросы для самоконтроля

.....

Курсовая работа — часть 2: Матричная математика и работа с пакетами

Для первоначального знакомства с пакетом рекомендую прочитать небольшой ман⁵ (англ.). Те, у кого плохо с английским, а это очень плохо для людей из мира информационных технологий, могут обратиться к Habr⁶, где есть более-менее полный и грамотный перевод упомянутого мана⁷:

- **NumPy в Python. Часть 1**⁸. Введение, общая характеристика библиотеки. Установка. Массивы в Python/
- **NumPy в Python. Часть 2**⁹. Создание массивов и манипуляции с элементами.
- **NumPy в Python. Часть 3**¹⁰. Операторы сравнения и тестирования значений. Матричная и векторная математика.
- **NumPy в Python. Часть 4**¹¹. Математика многочленов. Статистика. Случайные числа. Модули SciPy.

Для ускорения процесса написания курса молодого бойца нет смысла «толочь воду в ступе», используя собственное стилевое решение при редактировании еще одного варианта перевода известных основ. Частично материалы упомянутого выше перевода использованы в качестве базы фрагментов текста настоящих указаний, однако, в настолько переработанном виде, что это уже нельзя назвать цитированием

Итак,

⁵ <https://engineering.ucsb.edu/~shell/che210d/numpy.pdf>

⁶ <https://habr.com/>

⁷ Ман (проф. диалект) — от *manual* (англ.), руководство

⁸ <https://habr.com/ru/post/352678/>

⁹ <https://habr.com/ru/post/353416/>

¹⁰ <https://habr.com/ru/post/413381/>

¹¹ <https://habr.com/ru/post/415373/>

Кто такой этот NumPy?

NumPy — это open-source модуль для python, который предоставляет общие математические и числовые операции в виде пре-скомпилированных, быстрых функций. Они объединяются в высокоуровневые пакеты. Они обеспечивают функционал, который можно сравнить с функционалом MatLab. NumPy (Numeric Python) предоставляет базовые методы для манипуляции с большими массивами и матрицами. SciPy (Scientific Python) расширяет функционал numpy огромной коллекцией полезных алгоритмов, таких как минимизация, преобразование Фурье, регрессия и другие прикладные математические техники.

До сих пор, решая задачи «Разминки», мы не использовали пакеты расширений и особой настройки наша основная инструментальная среда, а это PyCharm, не требовала дополнительных настроек. Теперь нам потребуется, как минимум, установить нужные пакеты.

Можно воспользоваться оригинальной документацией, Имеется подробный «Путеводитель по NumPy»¹², который занимает 371 страницу английского текста. Имеется официальный сайт библиотеки¹³, которым, при должном усердии, вы будете пользоваться постоянно. Но для PyCharm разберёмся с этим вопросом подробнее.

¹² <http://web.mit.edu/dvp/Public/numpybook.pdf>

¹³ <http://www.numpy.org>

Настройка PyCharm и подключение библиотек NumPy и Mathplotlib

Вспомним, как создаётся новый проект в PyCharm, см. Рис. 89-49.

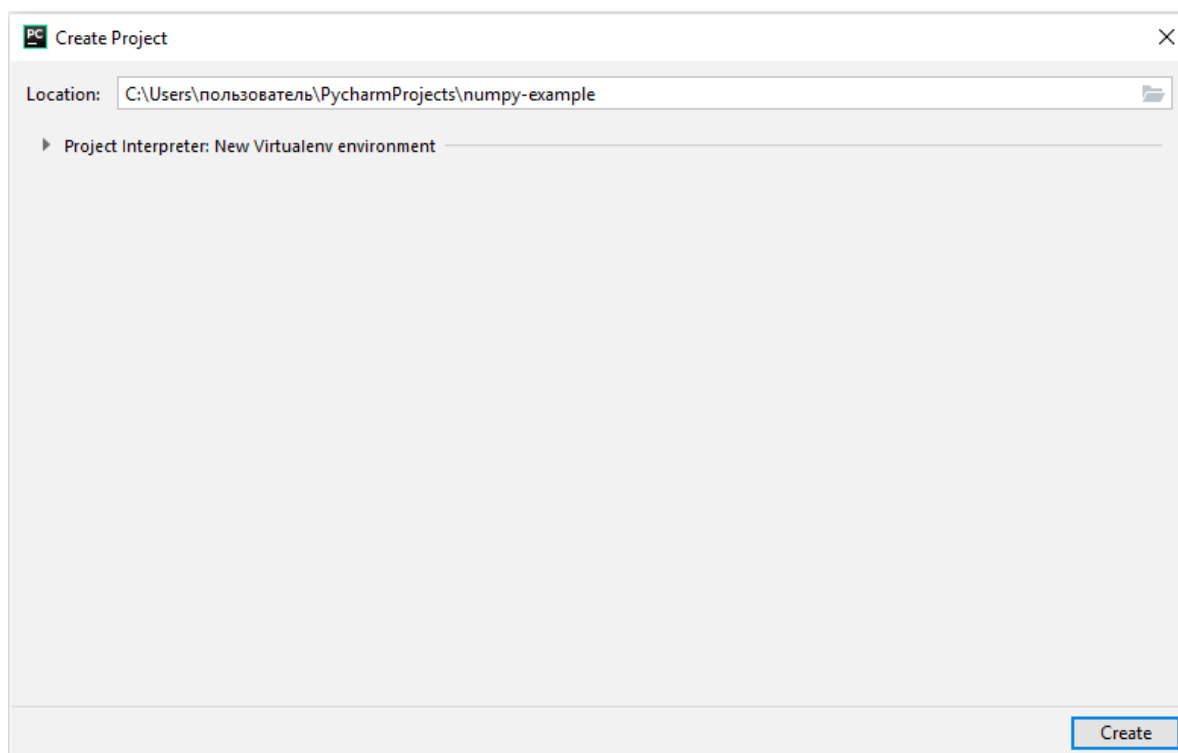
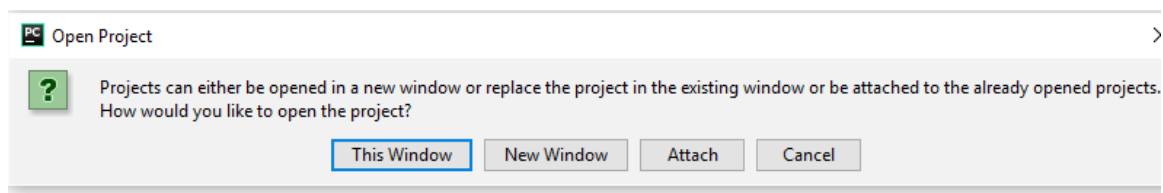


Рис. 90. Создаём и именуем новый проект



*Рис. 91. Новый проект размещаем в текущем окне PyCharm. Не нравится —
поступайте по-своему*

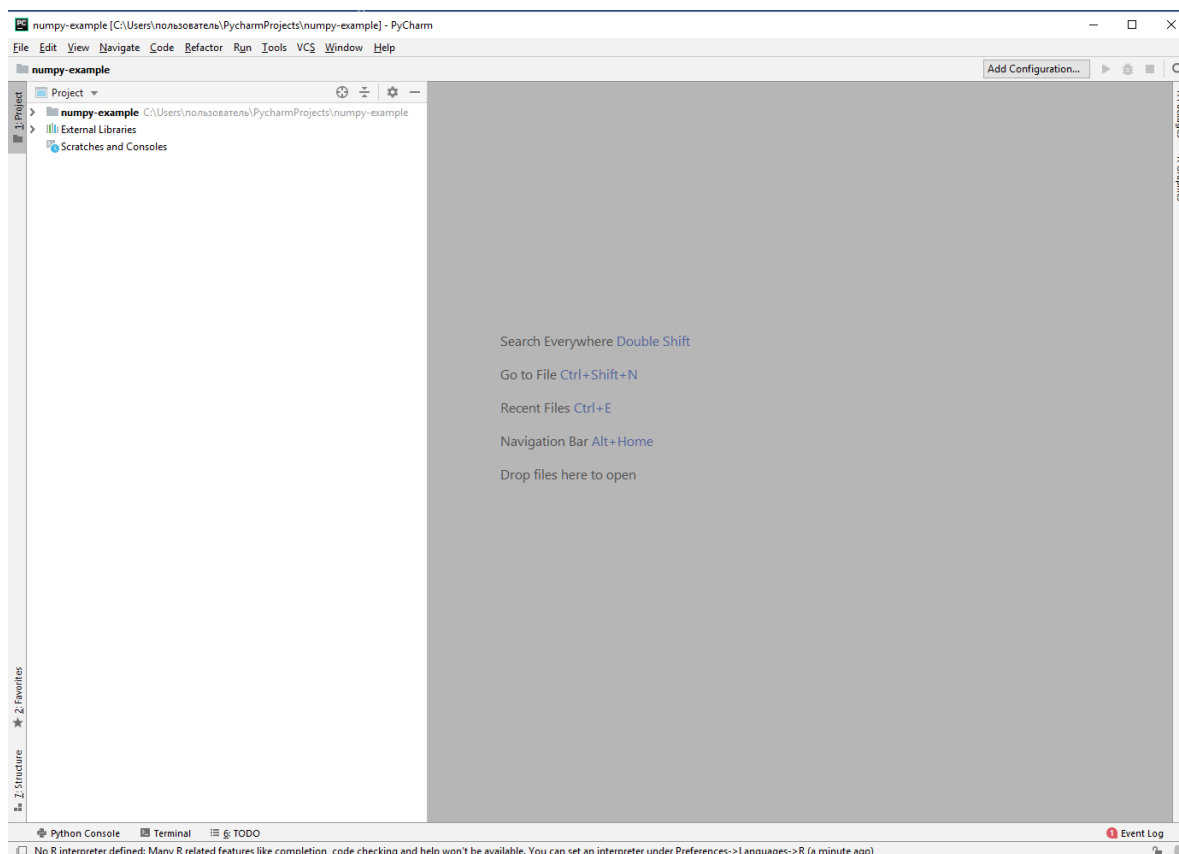


Рис. 92. Новый проект создан, приступаем к реализации

Первое, что мы сделаем, создадим новый проект, куда будем записывать все коды программ. Думаю, что особых трудностей это не вызовет см. рис. 47, 48 и 49.

Далее приступим к настройкам окружения для решения последующих задач, см. рис. 50, 51, 52, 53 и 54, как на путеводитель в интерфейсе — «делай раз, дела два, делай три ...».

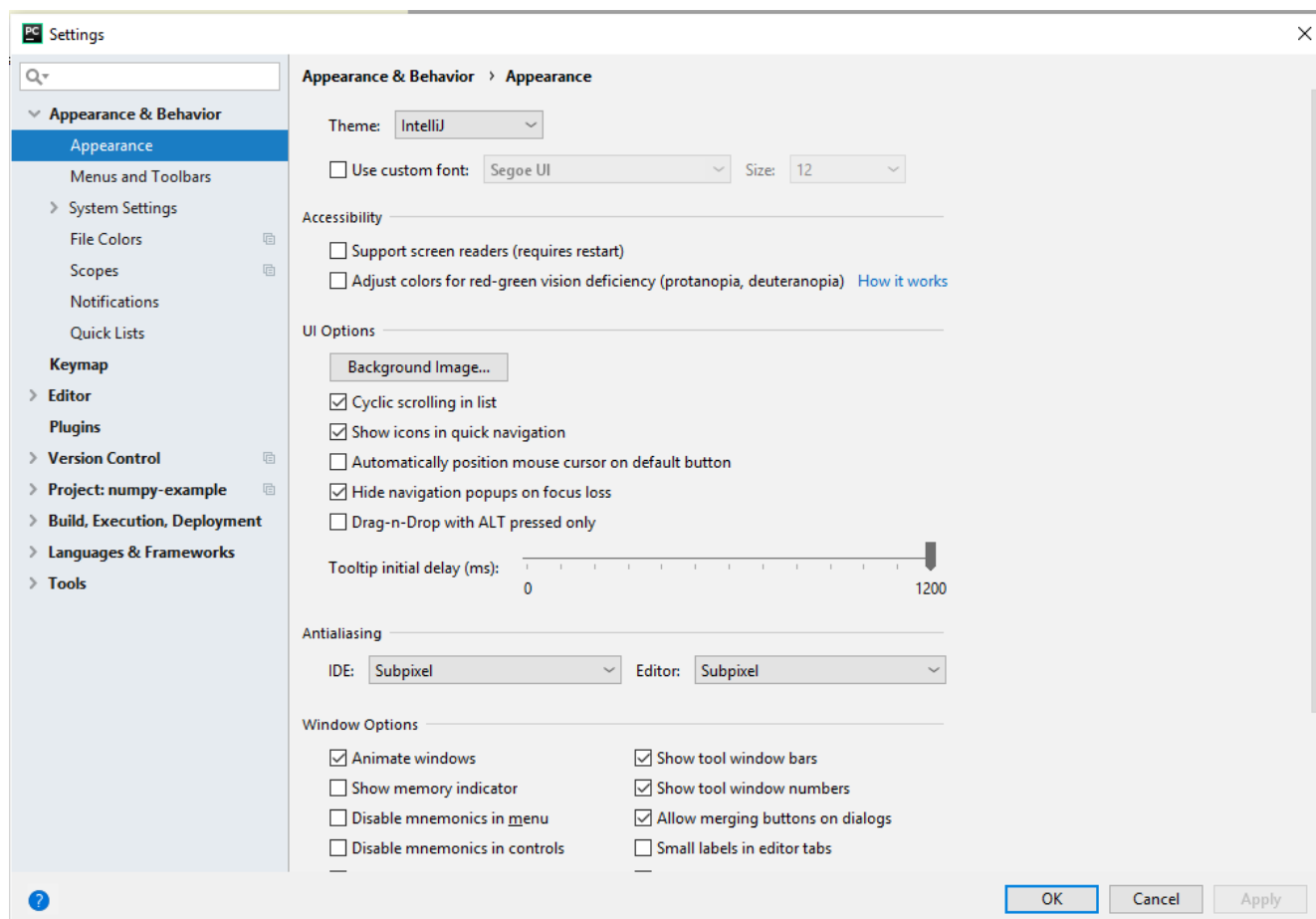


Рис. 93. Настройка окружения проекта

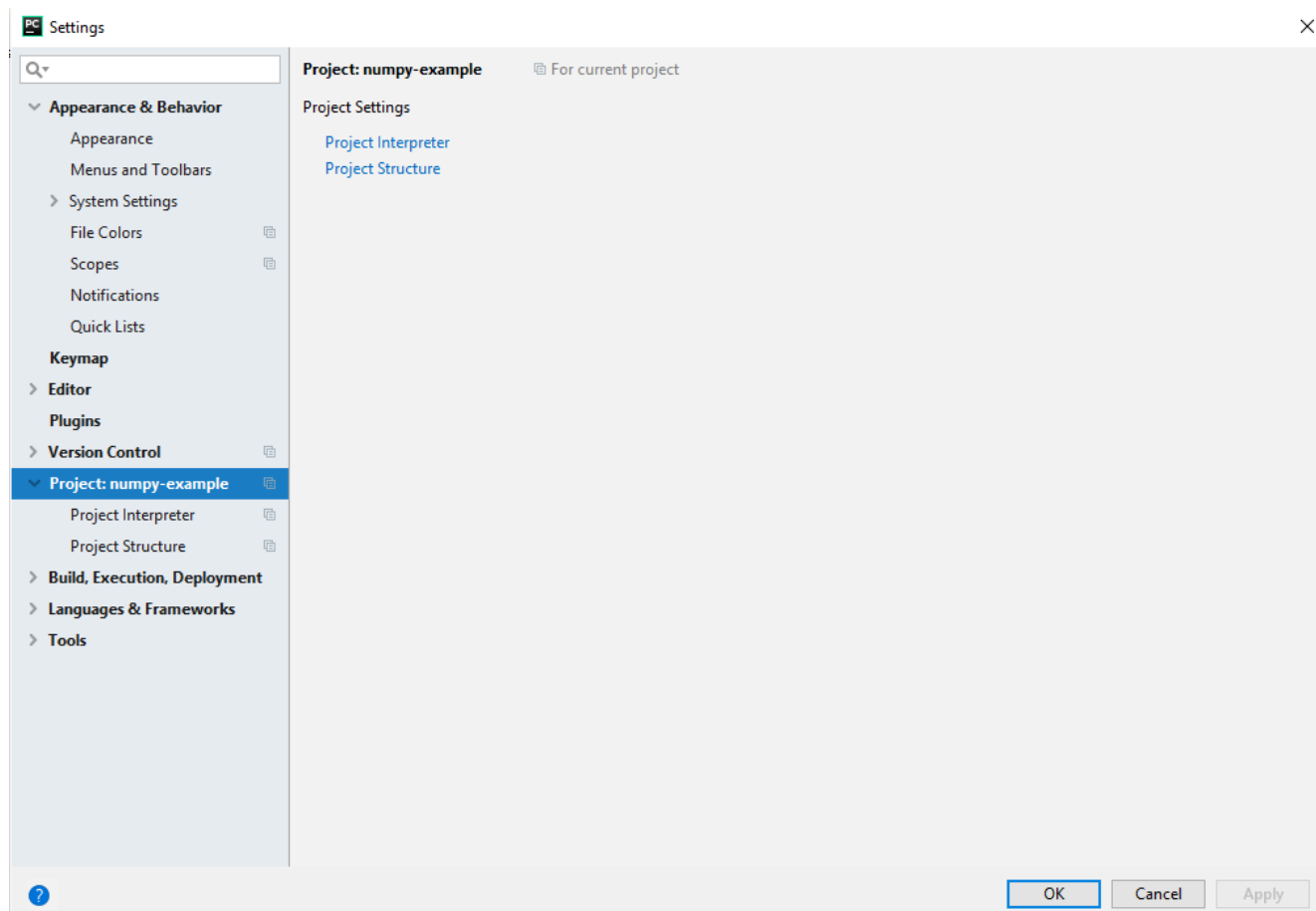


Рис. 94. Очень важно выбрать интерпретатор Project Interpreter

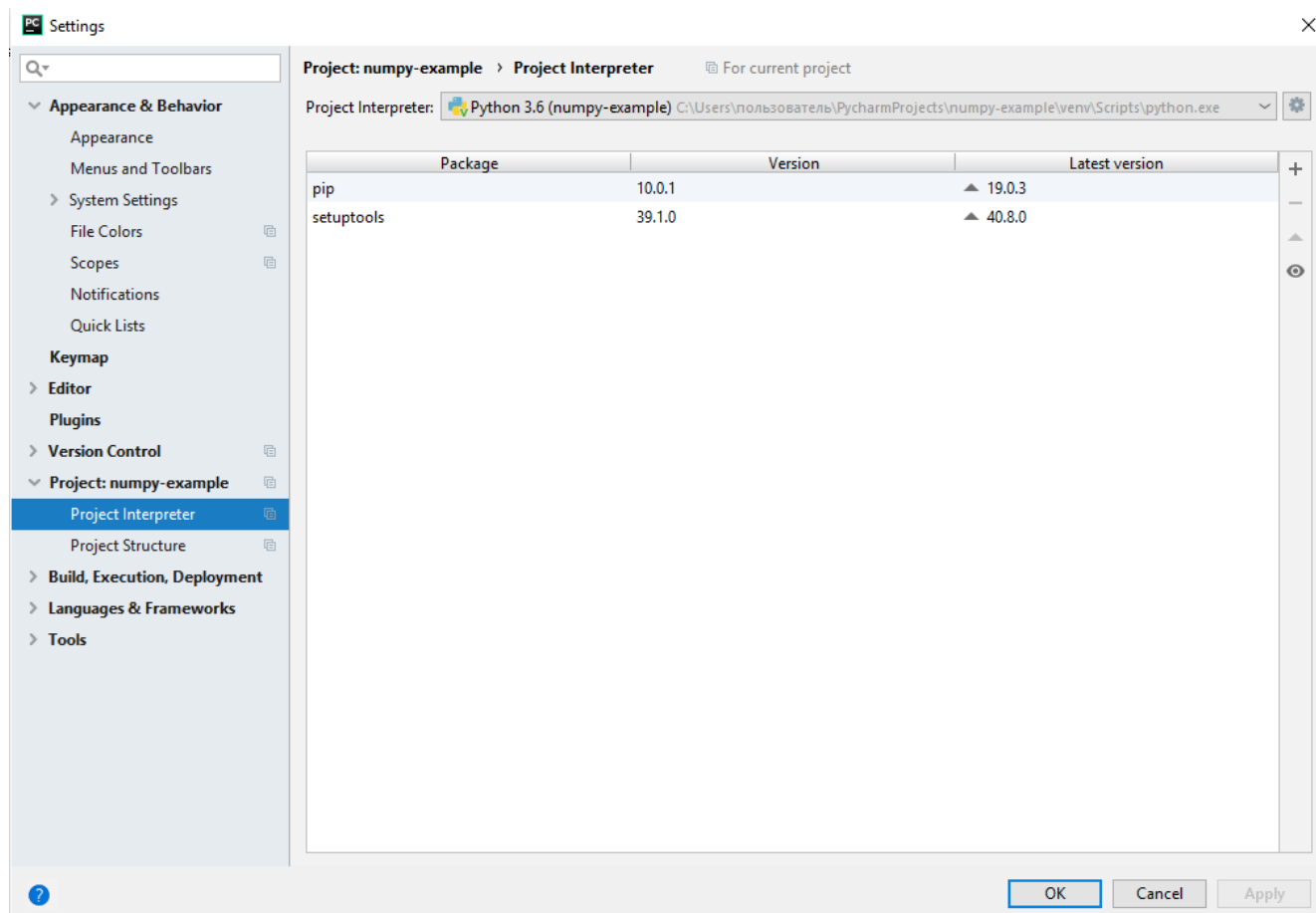


Рис. 95. Выбираем интерпретатор и добавляем пакеты — посмотрите на «+»
справа вверху

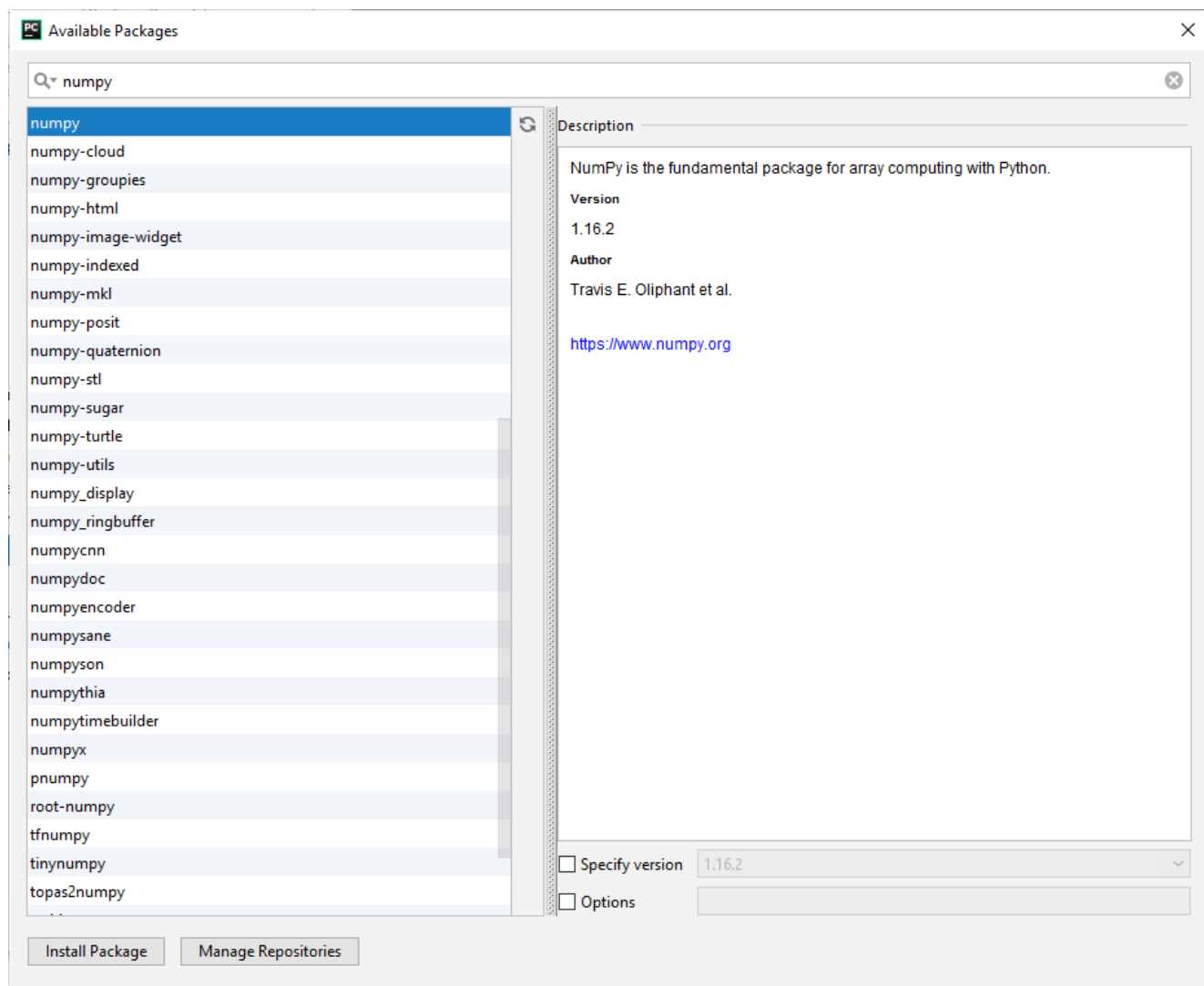


Рис. 96. Ищем numpy в репозитории и как только он будет найден, смело жмём кнопку «Install Package» для начала процесса установки пакета в окружение нашего проекта

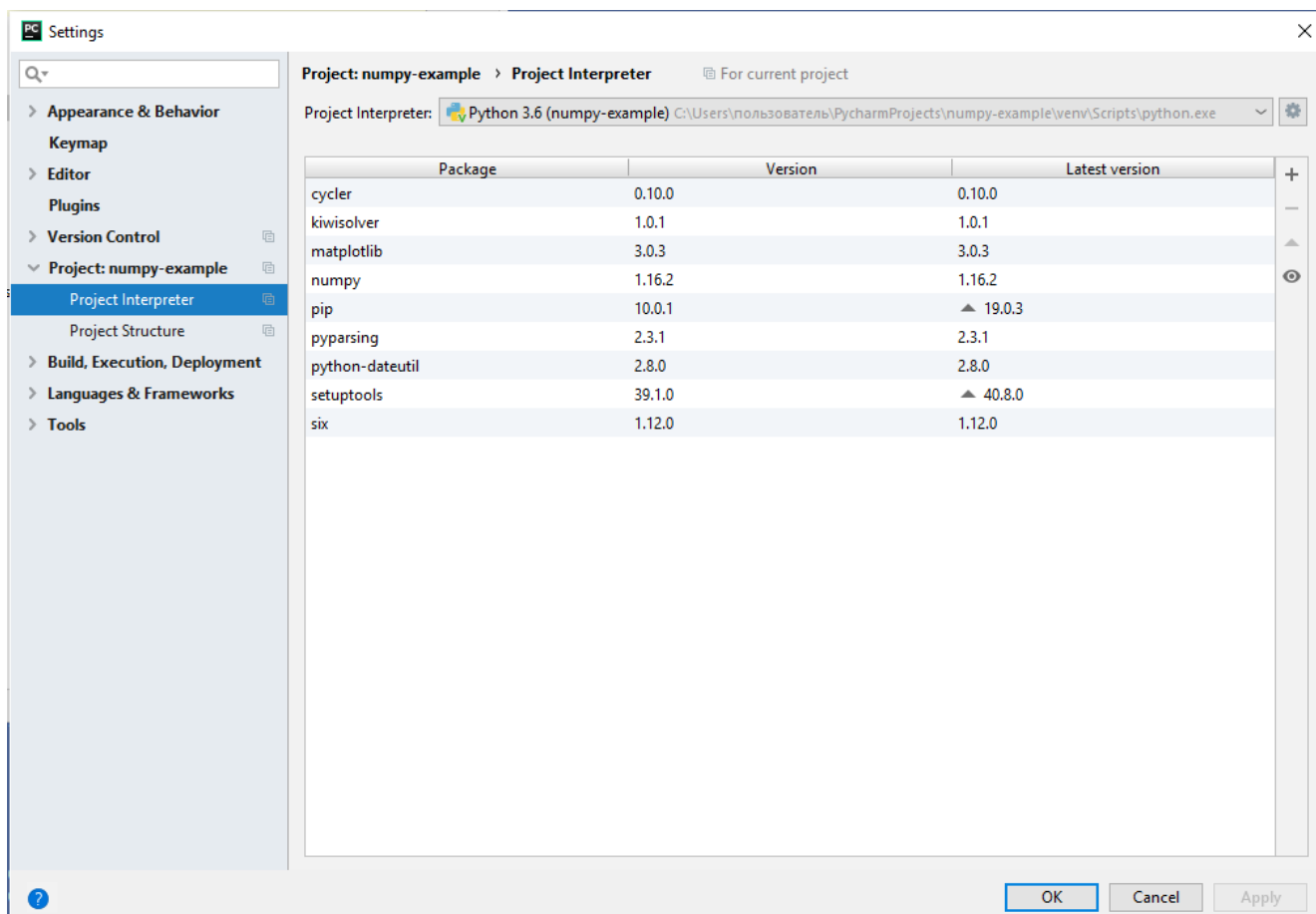


Рис. 97. Убедимся, что нужные пакеты есть в списке и установлены

Если всё сделали правильно, то можно приступить к записи кода и проверке работы пакетов. Ниже приведены примеры, которые можно просто скопировать в отдельные файлы, запустить и посмотреть, как это работает.

Примеры использования библиотек NumPy и Matplotlib

Код примеров, приведённых ниже, базируется на кодах, описанных на ресурсе «Записки программиста»¹⁴, но значительно переработаны, проверены и всё как положено. Исходные тексты программ можно получить по адресу

¹⁴ <https://eax.me/python-numpy/>

Итак, начинаем — создаем новый файл Python, в который записываем программный код первого примера (рис. 55).

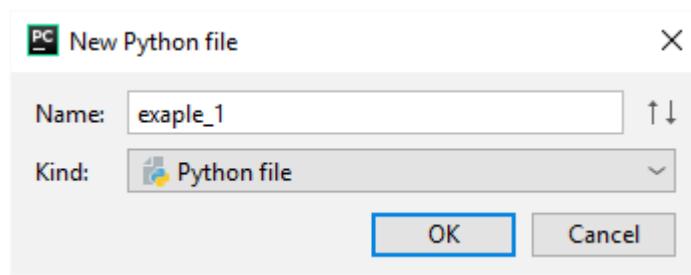


Рис. 98. Добавляем в проект *numpy-example* новый файл *example_1.py*
«Визуализация функций»

Код примера «Визуализация функций»

#имя проекта: numpy-example

#имя файла: example_1.py

#номер версии: 1.0¹⁵

#автор и его учебная группа: Е. Волков, ЭУ-142

#дата создания: 20.03.2019

дата последней модификации: 25.03.2019

#связанные файлы: пакеты numpy, matplotlib

описание: построение графика сигма-функции

#версия Python: 3.6

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-5, 5, 100)

def sigmoid(alpha):
    return 1 / ( 1 + np.exp(- alpha * x))
```

¹⁵ [Версия состоит из нескольких чисел \(как правило, трёх\), разделённых точкой: например, 1.5.2. Первое из них — старшая версия \(major\), второе — младшая \(minor\), третья — мелкие изменения \(maintenance, micro\). Для более подробной информации пройдите по этой ссылке и прочитайте в Wikipedia](#)

```

dpi = 80
fig = plt.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi))

plt.plot(x, sigmoid(0.5), 'ro-')
plt.plot(x, sigmoid(1.0), 'go-')
plt.plot(x, sigmoid(2.0), 'bo-')

plt.legend(['A = 0.5', 'A = 1.0', 'A = 2.0'], loc = 'upper left')
plt.show()

fig.savefig('sigmoid.png')

```

Результат примера «Визуализация функций»

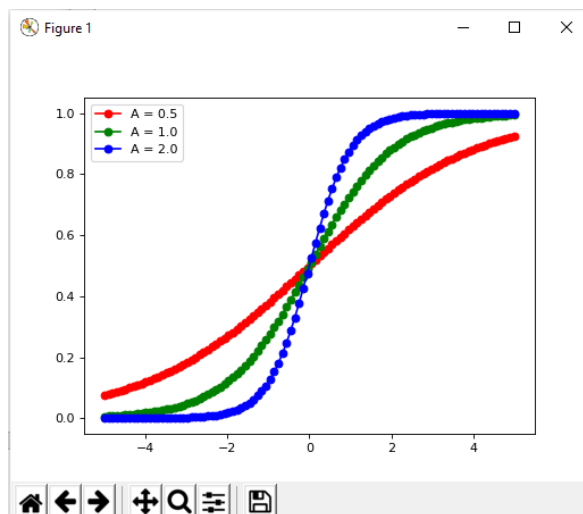


Рис. 99. Наблюдаем сигма-функцию

Код примера «Статистика»

#имя проекта: питру-example

#номер версии: 1.0

#имя файла: example_2.py

#автор и его учебная группа: Е. Волков, ЭУ-142

#дата создания: 20.03.2019

дата последней модификации: 25.03.2019

#связанные файлы: пакеты питру, matplotlib

описание: простейшие статистические вычисления

#версия Python: 3.6

```
ones = np.ones(50)
rnd = np.random.random(50) * 0.1
samples = ones + rnd
# Посчитаем среднее:
np.average(samples)
np.mean(samples)

# Медиану:

np.median(samples)

# Процентили:

np.percentile(samples, 50)
np.percentile(samples, 95)
np.percentile(samples, 99)

# Максимум, минимум, peak-to-peak:

samples.max()
samples.min()
samples.ptp()

# А заодно уж и стандартное отклонение с дисперсией:

np.std(samples)
np.var(samples)

# Используемая выше функция np.random.random генерирует случайные числа с
равномерным распределением. А если мы хотели бы использовать нормальное
распределение? Нет проблем:
#
import matplotlib.pyplot as plt
samples = np.random.normal(loc=0, scale=5, size=100000)
plt.hist(samples, 200)
plt.show()
```

Результат примера «Статистика»

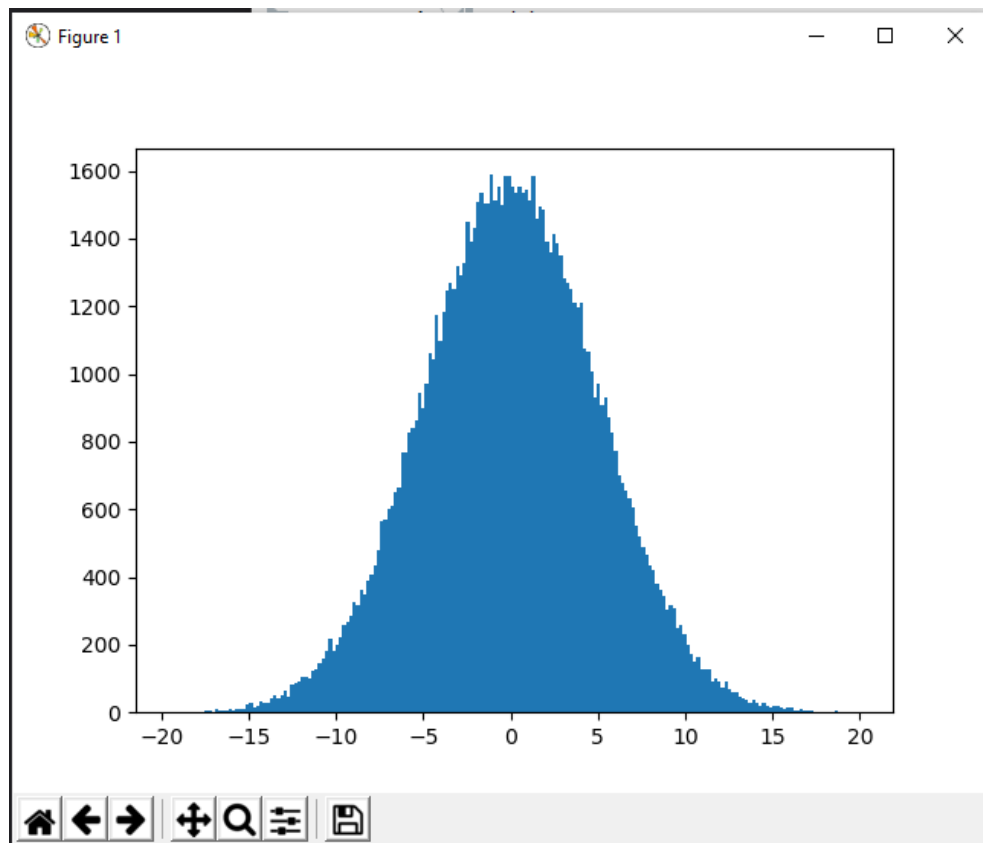


Рис. 100. Функция плотности вероятности случайной величины с нормальным законом распределения

Код примера «Линейная алгебра»

#имя проекта: питру-example

#номер версии: 1.0

#имя файла: example_3.py

#автор и его учебная группа: Е. Волков, ЭУ-142

#дата создания: 20.03.2019

дата последней модификации: 25.03.2019

#связанные файлы: пакеты питру, matplotlib

описание: линейная алгебра

#версия Python: 3.6

```
import numpy as np
```

```
import numpy.matlib
```

```

import numpy.linalg

m1 = np.arange(1, 10).reshape(3,3)
print(m1)

m2 = np.identity(3)
print(m2)
# Транспонируем первую матрицу, а также посчитаем след и детерминант второй:

m1.transpose()
print(m1)

m2.trace()
print(m2)

det = np.linalg.det(m2)
print(det)

# Матрицы можно складывать, умножать на число, умножать на вектор, а также
умножать на другую матрицу:

print(m1 + m2)

print(m1 * 3)

m1 + np.array([1,2,3])
print(m1 * m2)

# Посчитать матрицу, обратную к данной, можно функцией np.linalg.inv:
m3 = np.matlib.rand(3, 3)
(m3 * np.linalg.inv(m3))
print(m3)

print((m3 * np.linalg.inv(m3)).round())

```

Результат примера «Линейная алгебра»

```

[[1 2 3]
 [4 5 6]
 [7 8 9]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
[[1 2 3]
 [4 5 6]
 [7 8 9]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
1.0
[[ 2.  2.  3.]
 [ 4.  6.  6.]
 [ 7.  8. 10.]]
[[ 3 6 9]
 [12 15 18]]

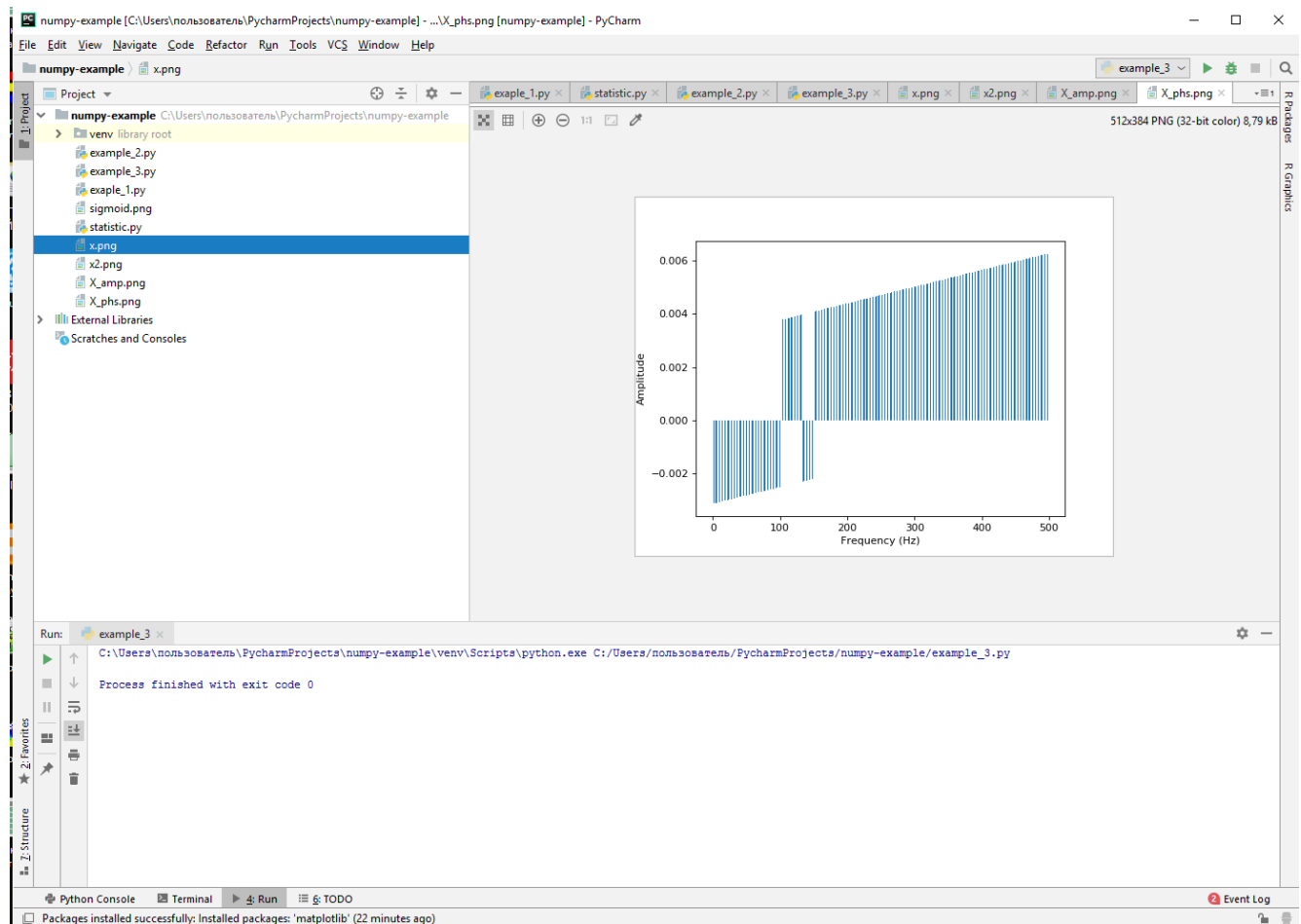
```

```

[21 24 27]]
[[1. 0. 0.]
 [0. 5. 0.]
 [0. 0. 9.]]
[[0.21960198 0.79353093 0.71379042]
 [0.79998886 0.24333293 0.4619364 ]
 [0.08058218 0.6665476 0.38367675]]
[[ 1. -0. -0.]
 [-0. 1. 0.]
 [ 0. 0. 1.]]

```

Process finished with exit code 0



Содержание части 2 курсовой работы

Для начала понимания, как работать с матрицами решите следующие задачи, количеством 31. Их можно записать в одном проекте.

Работа с двумерными массивами (матрицами)

1. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти наибольший элемент столбца матрицы A , для которого сумма абсолютных значений элементов максимальна.
2. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти наибольшее значение среди средних значений для каждой строки матрицы.
3. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти наименьший элемент столбца матрицы A , для которого сумма абсолютных значений элементов максимальна.
4. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти наименьшее значение среди средних значений для каждой строки матрицы.
5. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Определить средние значения по всем строкам и столбцам матрицы. Результат оформить в виде матрицы из $N + 1$ строк и $M + 1$ столбцов.
6. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти сумму элементов всей матрицы. Определить, какую долю в этой сумме составляет сумма элементов каждого столбца. Результат оформить в виде матрицы из $N + 1$ строк и M столбцов.
7. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти сумму элементов всей матрицы. Определить, какую долю в этой сумме составляет сумма элементов каждой строки. Результат оформить в виде матрицы из N строк и $M + 1$ столбцов.
8. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Определить, сколько отрицательных элементов

содержится в каждом столбце и в каждой строке матрицы. Результат оформить в виде матрицы из $N + 1$ строк и $M + 1$ столбцов.

9. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Определить, сколько нулевых элементов содержится в верхних L строках матрицы и в левых K столбцах матрицы.
10. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Перемножить элементы каждого столбца матрицы с соответствующими элементами K -го столбца.
11. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Просуммировать элементы каждой строки матрицы с соответствующими элементами L -й строки.
12. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Разделить элементы каждой строки на элемент этой строки с наибольшим значением.
13. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Разделить элементы каждого столбца матрицы на элемент этого столбца с наибольшим значением.
14. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Разделить элементы матрицы на элемент матрицы с наибольшим значением.
15. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Все элементы имеют целый тип. Дано целое число H . Определить, какие столбцы имеют хотя бы одно такое число, а какие не имеют.
16. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Исключить из матрицы строку с номером L . Сомкнуть строки матрицы.

17. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Добавить к матрице строку и вставить ее под номером L .
18. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Найти сумму элементов, стоящих на главной диагонали, и сумму элементов, стоящих на побочной диагонали (элементы главной диагонали имеют индексы от $[0,0]$ до $[N,N]$, а элементы побочной диагонали — от $[N,0]$ до $[0,N]$).
19. Создать квадратную матрицу A , имеющую N строк и N столбцов со случайными элементами. Определить сумму элементов, расположенных параллельно главной диагонали (ближайшие к главной). Элементы главной диагонали имеют индексы от $[0,0]$ до $[N,N]$.
20. Создать квадратную матрицу A , имеющую N строк и N столбцов со случайными элементами. Определить произведение элементов, расположенных параллельно побочной диагонали (ближайшие к побочной). Элементы побочной диагонали имеют индексы от $[N,0]$ до $[0,N]$.
21. Создать квадратную матрицу A , имеющую N строк и N столбцов со случайными элементами. Каждой паре элементов, симметричных относительно главной диагонали (ближайшие к главной), присвоить значения, равные полусумме этих симметричных значений (элементы главной диагонали имеют индексы от $[0,0]$ до $[N,N]$).
22. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Исходная матрица состоит из нулей и единиц. Добавить к матрице еще один столбец, каждый элемент которого делает количество единиц в каждой строке чётным.
23. Создать квадратную матрицу A , имеющую N строк и N столбцов со случайными элементами. Найти сумму элементов, расположенных выше

главной диагонали, и произведение элементов, расположенных выше побочной диагонали (элементы главной диагонали имеют индексы от $[0,0]$ до $[N,N]$, а элементы побочной диагонали — от $[N,0]$ до $[0,N]$).

24. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Дан номер строки L и номер столбца K , при помощи которых исходная матрица разбивается на четыре части. Найти сумму элементов каждой части.
25. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Определить, сколько нулевых элементов содержится в каждом столбце и в каждой строке матрицы. Результат оформить в виде матрицы из $N + 1$ строк и $M + 1$ столбцов.
26. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Дан номер строки L и номер столбца K , при помощи которых исходная матрица разбивается на четыре части. Найти среднее арифметическое элементов каждой части.
27. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Все элементы имеют целый тип. Дано целое число N . Определить, какие строки имеют хотя бы одно такое число, а какие не имеют.
28. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Исключить из матрицы столбец с номером K . Сомкнуть столбцы матрицы.
29. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Добавить к матрице столбец чисел и вставить его под номером K .
30. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Добавить к элементам каждого столбца такой новый элемент, чтобы сумма положительных элементов стала бы равна

модулю суммы отрицательных элементов. Результат оформить в виде матрицы из $N + 1$ строк и M столбцов.

31. Создать прямоугольную матрицу A , имеющую N строк и M столбцов со случайными элементами. Добавить к элементам каждой строки такой новый элемент, чтобы сумма положительных элементов стала бы равна модулю суммы отрицательных элементов. Результат оформить в виде матрицы из N строк и $M + 1$ столбцов.

Решение СЛАУ методом Гаусса — немного теории

Метод Гаусса при решении СЛАУ¹⁶ позволяет ответить на вопросы о совместности или несовместности, определенности или неопределенности системы линейных уравнений, а также отыскать все решения совместной системы.

В основе метода лежит идея последовательного исключения неизвестных с помощью подстановок, суть которой состоит в приведении данной системы к другой, равносильной ей, но более простой системе. Это приведение одной системы к другой осуществляется путем элементарных преобразований, которые производятся над уравнениями системы или, что удобнее, над строками расширенной матрицы. Элементарному преобразованию системы линейных уравнений соответствует одноименное элементарное преобразование строк ее расширенной матрицы.

¹⁶ СЛАУ – система линейных алгебраических уравнений

СУТЬ И АЛГОРИТМ РЕШЕНИЕ СЛАУ МЕТОДОМ ГАУССА — ПРИМЕР:

$$\begin{cases} x_1 + 2x_2 + 3x_3 - 2x_4 = 6 \\ 2x_1 - x_2 - 2x_3 - 3x_4 = 8 \\ 3x_1 + 2x_2 - x_3 + 2x_4 = 4 \\ 2x_1 - 3x_2 + 2x_3 + x_4 = -8 \end{cases}.$$

РЕШЕНИЕ:

Выпишем расширенную матрицу **B** данной системы и приведем ее к ступенчатому виду:

$$B = \left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 2 & -1 & -2 & -3 & 8 \\ 3 & 2 & -1 & 2 & 4 \\ 2 & -3 & 2 & 1 & -8 \end{array} \right).$$

Последовательно умножим первую строку на (-2) и прибавим ее ко второй строке, затем умножим на (-3) и прибавим к третьей строке, умножим на (-2) и прибавим к четвертой строке, получим

$$B \rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 0 & -5 & -8 & 1 & -4 \\ 0 & -4 & -10 & 8 & -14 \\ 0 & -7 & -4 & 5 & -20 \end{array} \right).$$

Ко второй строке полученной матрицы прибавим третью строку, умноженную на (-1) , затем во вновь полученной матрице умножим третью строку на $(-1/2)$, четвертую — на (-1) , затем последовательно умножим вторую строку на 2 и прибавим ее к третьей строке, умножим на 7 и прибавим к четвертой строке, получим

$$\left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 0 & -1 & 2 & -7 & 10 \\ 0 & 2 & 5 & -4 & 7 \\ 0 & 7 & 4 & -5 & 20 \end{array}\right) \rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 0 & -1 & 2 & -7 & 10 \\ 0 & 0 & 9 & -18 & 27 \\ 0 & 0 & 18 & -54 & 90 \end{array}\right).$$

Третью строку полученной матрицы умножим на **1/9**, четвертую — на **1/18**, затем третью строку умножим на **(-1)** и прибавим к четвертой строке, получим

$$\left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 0 & -1 & 2 & -7 & 10 \\ 0 & 0 & 1 & -2 & 3 \\ 0 & 0 & 1 & -3 & 5 \end{array}\right) \rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 3 & -2 & 6 \\ 0 & -1 & 2 & -7 & 10 \\ 0 & 0 & 1 & -2 & 3 \\ 0 & 0 & 0 & -1 & 2 \end{array}\right).$$

Найденная матрица имеет треугольный вид; по этой матрице запишем систему уравнений, эквивалентную исходной системе,

$$\begin{cases} x_1 + 2x_2 + 3x_3 - 2x_4 = 6 \\ -x_2 + 2x_3 - 7x_4 = 10 \\ x_3 - 2x_4 = 3 \\ -x_4 = 2 \end{cases}.$$

1. Последовательно находим неизвестные, начиная с последнего уравнения, $x_4 = -2$;
2. подставим x_4 в третье уравнение и найденное x_3 , $x_3 = -1$;
3. затем из второго уравнения находим $x_2 = 2$;
4. из первого уравнения получим $x_1 = 1$.

Ответ: **$x_1 = 1, x_2 = 2, x_3 = -1, x_4 = -2$** .

Код программы решения СЛАУ методом Гаусса

```
#имя проекта: numru-example
#номер версии: 1.0
#имя файла: gauss.py
```

```

#автор и его учебная группа: Е. Волков, ЭУ-142
#дата создания: 20.03.2019
#дата последней модификации: 25.03.2019
#связанные файлы: нет
#описание: решение СЛАУ методом Гаусса
#версия Python: 3.6

# --- исходные данные
myA = [
    [1.0, -2.0, 3.0, -4.0],
    [3.0, 3.0, -5.0, -1.0],
    [3.0, 0.0, 3.0, -10.0],
    [-2.0, 1.0, 2.0, -3.0]
]

myB = [
    2.0,
    -3.0,
    8.0,
    5.0]
# --- end of исходные данные

# --- вывод системы на экран
def FancyPrint(A, B, selected):
    for row in range(len(B)):
        print("(", end='')
        for col in range(len(A[row])):
            print("\t{1:10.2f}{0}".format(" " if (selected is None
or selected != (row, col)) else "*", A[row][col]),
                end='')
            print("\t) * (\tX{0}) = (\t{1:10.2f})".format(row + 1,
B[row]))
# --- end of вывод системы на экран

# --- перемена местами двух строк системы
def SwapRows(A, B, row1, row2):
    A[row1], A[row2] = A[row2], A[row1]
    B[row1], B[row2] = B[row2], B[row1]

# --- end of перемена местами двух строк системы

```

```

# --- деление строки системы на число
def DivideRow(A, B, row, divider):
    A[row] = [a / divider for a in A[row]]
    B[row] /= divider

# --- end of деление строки системы на число

# --- сложение строки системы с другой строкой, умноженной на число
def CombineRows(A, B, row, source_row, weight):
    A[row] = [(a + k * weight) for a, k in zip(A[row],
A[source_row])]
    B[row] += B[source_row] * weight

# --- end of сложение строки системы с другой строкой, умноженной на
число

# --- решение системы методом Гаусса (приведением к треугольному
виду)
def Gauss(A, B):
    column = 0
    while (column < len(B)):

        print("Ищем максимальный по модулю элемент в {0}-м
столбце:".format(column + 1))
        current_row = None
        for r in range(column, len(A)):
            if current_row is None or abs(A[r][column]) >
abs(A[current_row][column]):
                current_row = r
        if current_row is None:
            print("решений нет")
            return None
        FancyPrint(A, B, (current_row, column))

        if current_row != column:
            print("Переставляем строку с найденным элементом
повыше:")

```

```

        SwapRows(A, B, current_row, column)
        FancyPrint(A, B, (column, column))

    print("Нормализуем строку с найденным элементом:")
    DivideRow(A, B, column, A[column][column])
    FancyPrint(A, B, (column, column))

    print("Обрабатываем нижележащие строки:")
    for r in range(column + 1, len(A)):
        CombineRows(A, B, r, column, -A[r][column])
        FancyPrint(A, B, (column, column))

    column += 1

    print("Матрица приведена к треугольному виду, считаем решение")
    X = [0 for b in B]
    for i in range(len(B) - 1, -1, -1):
        X[i] = B[i] - sum(x * a for x, a in zip(X[(i + 1):], A[i][(i
+ 1):]))

    print("Получили ответ:")
    print("\n".join("X{0} =\t{1:10.2f}".format(i + 1, x) for i, x in
enumerate(X)))

    return X

# --- end of решение системы методом Гаусса (приведением к
треугольному виду)

print("Исходная система:")
FancyPrint(myA, myB, None)

print("Решаем:")
Gauss(myA, myB)

```

Результат работы программы для решения СЛА методом Гаусса

```

C:\Users\пользователь\PycharmProjects\numpy-
example\venv\Scripts\python.exe
C:/Users/пользователь/PycharmProjects/numpy-example/gauss.py
Исходная система:
( 1.00 -2.00 3.00 -4.00 ) * ( X1 ) = ( 2.00)

```

$$\begin{pmatrix} 3.00 & 3.00 & -5.00 & -1.00 \end{pmatrix} * (X2) = (-3.00)$$

$$\begin{pmatrix} 3.00 & 0.00 & 3.00 & -10.00 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} -2.00 & 1.00 & 2.00 & -3.00 \end{pmatrix} * (X4) = (5.00)$$

Решаем:

Ищем максимальный по модулю элемент в 1-м столбце:

$$\begin{pmatrix} 1.00 & -2.00 & 3.00 & -4.00 \end{pmatrix} * (X1) = (2.00)$$

$$\begin{pmatrix} 3.00^* & 3.00 & -5.00 & -1.00 \end{pmatrix} * (X2) = (-3.00)$$

$$\begin{pmatrix} 3.00 & 0.00 & 3.00 & -10.00 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} -2.00 & 1.00 & 2.00 & -3.00 \end{pmatrix} * (X4) = (5.00)$$

Переставляем строку с найденным элементом повыше:

$$\begin{pmatrix} 3.00^* & 3.00 & -5.00 & -1.00 \end{pmatrix} * (X1) = (-3.00)$$

$$\begin{pmatrix} 1.00 & -2.00 & 3.00 & -4.00 \end{pmatrix} * (X2) = (2.00)$$

$$\begin{pmatrix} 3.00 & 0.00 & 3.00 & -10.00 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} -2.00 & 1.00 & 2.00 & -3.00 \end{pmatrix} * (X4) = (5.00)$$

Нормализуем строку с найденным элементом:

$$\begin{pmatrix} 1.00^* & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} 1.00 & -2.00 & 3.00 & -4.00 \end{pmatrix} * (X2) = (2.00)$$

$$\begin{pmatrix} 3.00 & 0.00 & 3.00 & -10.00 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} -2.00 & 1.00 & 2.00 & -3.00 \end{pmatrix} * (X4) = (5.00)$$

Обрабатываем нижележащие строки:

$$\begin{pmatrix} 1.00^* & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} 0.00 & -3.00 & 4.67 & -3.67 \end{pmatrix} * (X2) = (3.00)$$

$$\begin{pmatrix} 0.00 & -3.00 & 8.00 & -9.00 \end{pmatrix} * (X3) = (11.00)$$

$$\begin{pmatrix} 0.00 & 3.00 & -1.33 & -3.67 \end{pmatrix} * (X4) = (3.00)$$

Ищем максимальный по модулю элемент в 2-м столбце:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} 0.00 & -3.00^* & 4.67 & -3.67 \end{pmatrix} * (X2) = (3.00)$$

$$\begin{pmatrix} 0.00 & -3.00 & 8.00 & -9.00 \end{pmatrix} * (X3) = (11.00)$$

$$\begin{pmatrix} 0.00 & 3.00 & -1.33 & -3.67 \end{pmatrix} * (X4) = (3.00)$$

Нормализуем строку с найденным элементом:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} -0.00 & 1.00^* & -1.56 & 1.22 \end{pmatrix} * (X2) = (-1.00)$$

$$\begin{pmatrix} 0.00 & -3.00 & 8.00 & -9.00 \end{pmatrix} * (X3) = (11.00)$$

$$\begin{pmatrix} 0.00 & 3.00 & -1.33 & -3.67 \end{pmatrix} * (X4) = (3.00)$$

Обрабатываем нижележащие строки:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} -0.00 & 1.00^* & -1.56 & 1.22 \end{pmatrix} * (X2) = (-1.00)$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33 & -5.33 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33 & -7.33 \end{pmatrix} * (X4) = (6.00)$$

Ищем максимальный по модулю элемент в 3-м столбце:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * (X1) = (-1.00)$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * (X2) = (-1.00)$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33 & -5.33 \end{pmatrix} * (X3) = (8.00)$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33^* & -7.33 \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 6.00 \end{pmatrix}$$

Переставляем строку с найденным элементом повыше:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33^* & -7.33 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 6.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33 & -5.33 \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 8.00 \end{pmatrix}$$

Нормализуем строку с найденным элементом:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 1.00^* & -2.20 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 1.80 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 3.33 & -5.33 \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 8.00 \end{pmatrix}$$

Обрабатываем нижележащие строки:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 1.00^* & -2.20 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 1.80 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 0.00 & 2.00 \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 2.00 \end{pmatrix}$$

Ищем максимальный по модулю элемент в 4-м столбце:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 1.00 & -2.20 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 1.80 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 0.00 & 2.00^* \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 2.00 \end{pmatrix}$$

Нормализуем строку с найденным элементом:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 1.00 & -2.20 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 1.80 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 0.00 & 1.00^* \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 1.00 \end{pmatrix}$$

Обрабатываем нижележащие строки:

$$\begin{pmatrix} 1.00 & 1.00 & -1.67 & -0.33 \end{pmatrix} * \begin{pmatrix} X1 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} -0.00 & 1.00 & -1.56 & 1.22 \end{pmatrix} * \begin{pmatrix} X2 \end{pmatrix} = \begin{pmatrix} -1.00 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 1.00 & -2.20 \end{pmatrix} * \begin{pmatrix} X3 \end{pmatrix} = \begin{pmatrix} 1.80 \end{pmatrix}$$

$$\begin{pmatrix} 0.00 & 0.00 & 0.00 & 1.00^* \end{pmatrix} * \begin{pmatrix} X4 \end{pmatrix} = \begin{pmatrix} 1.00 \end{pmatrix}$$

Матрица приведена к треугольному виду, считаем решение

Получили ответ:

$$X1 = 2.00$$

$$X2 = 4.00$$

$$X3 = 4.00$$

$$X4 = 1.00$$

Process finished with exit code 0

Программа решения СЛАУ с использованием пакета numru

#имя проекта: numru-example

#номер версии: 1.0


```

#имя файла: numpy-gauss.py
#автор и его учебная группа: Е. Волков, ЭУ-142
#дата создания: 20.03.2019
# дата последней модификации: 25.03.2019
#связанные файлы: пакеты numpy
# описание: решение СЛАУ с использование библиотеки numpy
#версия Python: 3.6

import numpy # импортируем библиотеку

# --- исходные данные
myA = [
    [1.0, -2.0, 3.0, -4.0],
    [3.0, 3.0, -5.0, -1.0],
    [3.0, 0.0, 3.0, -10.0],
    [-2.0, 1.0, 2.0, -3.0]
]

myB = [
    2.0,
    -3.0,
    8.0,
    5.0]
# --- end of исходные данные

# --- вывод системы на экран
def FancyPrint(A, B, selected):
    for row in range(len(B)):
        print("(", end='')
        for col in range(len(A[row])):
            print("\t{1:10.2f}{0}".format(" " if (selected is None
or selected != (row, col)) else "*", A[row][col]),
                end='')
            print("\t) * (\tX{0}) = (\t{1:10.2f})".format(row + 1,
B[row]))
# --- end of вывод системы на экран

print("Исходная система:")
FancyPrint(myA, myB, None)

slv = numpy.linalg.solve(myA, myB)

```

```
print("Решаем:")  
print(slv)
```

Результат решения СЛАУ с использованием пакета numpy

```
C:\Users\пользователь\PycharmProjects\numpy-  
example\venv\Scripts\python.exe
```

```
C:/Users/пользователь/PycharmProjects/numpy-example/numpy-gauss.py
```

Исходная система:

```
( 1.00 -2.00 3.00 -4.00 ) * ( X1) = ( 2.00)  
( 3.00 3.00 -5.00 -1.00 ) * ( X2) = ( -3.00)  
( 3.00 0.00 3.00 -10.00 ) * ( X3) = ( 8.00)  
( -2.00 1.00 2.00 -3.00 ) * ( X4) = ( 5.00)
```

Решаем:

```
[2. 4. 4. 1.]
```

```
Process finished with exit code 0
```

Задание курсовой работы для решения СЛАУ методом Гаусса

Варианты задания находятся в приложении 5. Номер вашего индивидуального задания — это две последние цифры студенческого билета. Например, номер студенческого 301/08 — Ваш вариант 08.

В приведенных выше примерах исходные данные, квадратная матрица A и вектор B в правой части СЛАУ задаются операцией присваивания:

```
# --- исходные данные  
myA = [[1.0, -2.0, 3.0, -4.0],  
       [3.0, 3.0, -5.0, -1.0],  
       [3.0, 0.0, 3.0, -10.0],  
       [-2.0, 1.0, 2.0, -3.0]]  
myB = [2.0, -3.0, 8.0, 5.0]
```

```
# --- end of исходные данные
```

Ваша задача — записать в файл формата .csv (MS Excel) исходные для всех систем уравнений своего, применить к ним последовательно соответствующий метод решения и результаты вывести на экран и записать в файл.

Библиографический список

1. Вирт, Н. Алгоритмы и структуры данных / Н. Вирт. — М: Изд-во «Мир», 1989. — 360 с.;

Шпаргалка Python №1

Базовые типы

integer, float, boolean, string

`int` 783 0 -192

`float` 9.23 0.0 -1.7e-6

`bool` True False

`str` "One\nTwo" 'I\m'

↑
неизменяемая, упорядоченная последовательность символов

перевод строки
экранировка
многострочные
символ табуляции

Контейнерные типы

- упорядоченная последовательность, быстрый доступ по индексу
- выражение с одними запятыми
- как упорядоченная последовательность символов
- порядок заранее неизвестен, быстрый доступ по ключу, ключи = базовые типы или кортежи
- соответствие между ключами и значениями

`list` [1,5,9] ["x",11,8.9] ["word"] []

`tuple` (1,5,9) 11,"y",7.4 ("word",) ()

`str`

`dict` {"key":"value"} {}

`set` {"key1","key2"} {1,9,3,0} set()

Имена

для переменных, функций, модулей, классов...

`a..zA..Z`, потом `a..zA..Z_0..9`

- нелатинские буквы запрещены, но избегайте их
- ключевые слова языка запрещены
- маленькие/БОЛЬШИЕ буквы отличаются

• `a toto x7 y_max BigOne`

• `8y and`

Преобразования

`type` (выражение)

`int("15")` можно указать целое основание системы исчисления вторым параметром

`int(15.56)` отбросить дробную часть (для округления делайте `round(15.56)`)

`float("-11.24e8")`

`str(78.3)` и для буквального преобразования → `repr("Text")`

`bool` → используйте сравнения (`==`, `!=`, `<`, `>`, ...), дающие логический результат

`list("abc")` → использует каждый элемент последовательности → ['a', 'b', 'c']

`dict([(3,"three"), (1,"one")])` → {1:'one', 3:'three'}

`set(["one", "two"])` → использует каждый элемент последовательности → {'one', 'two'}

`":".join(['toto', '12', 'pswd'])` → 'toto:12:pswd'

соединяющая строка последовательность строк

`"words with spaces".split()` → ['words', 'with', 'spaces']

`"1,4,8,2".split(",")` → ['1', '4', '8', '2']

строка-разделитель

Присвоение переменным

`x = 1.2+8+sin(0)`

↑
значение или вычисляемое выражение

`y,z,r = 9.2,-7.6,"bad"`

имена переменных контейнер с несколькими значениями (здесь кортеж)

`x+=3` добавление вычитание → `x-=2`

`x=None` «неопределённая» константа

Доступ к элементам последовательностей

`len(lst)` → 6

доступ к отдельным элементам через [индекс]

`lst[1]→67` `lst[0]→11` первый

`lst[-2]→42` `lst[-1]→1968` последний

доступ к подпоследовательности [начало среза : конец среза : шаг]

`lst[1:3]→[67,"abc"]`

`lst[-3:-1]→[3.14,42]`

`lst[:3]→[11,67,"abc"]`

`lst[4:]→[42,1968]`

срез без указания границ → с начала до конца

Для изменяемых последовательностей, полезно удаление `del lst[3:5]` и изменение с помощью присвоения `lst[1:4]=['hop',9]`

Булева логика

Сравнения: `<` `>` `<=` `>=` `==` `!=`

`a and b` логическое и оба верны одновременно

`a or b` логическое или верно хотя бы одно

`not a` логическое нет

`True` константа «истина»

`False` константа «ложь»

Условный оператор

выражения в блоке выполняются только если условие истинно

`if` логическое выражение:

→ блок выражений

может сопровождаться несколькими `elif`, `elif`, ..., но только одним окончательным `else`. Пример:

`if x==42:`

блок выполнится, если x==42 истинно

`print("real truth")`

`elif x>0:`

иначе блок, если лог. выражение x > 0 истинно

`print("be positive")`

`elif bFinished:`

иначе блок, если лог. перем. bFinished истинна

`print("how, finished")`

`else:`

иначе блок для всех остальных случаев

`print("when it's not")`

Математика

числа с плавающей точкой... приближенные значения!

Операторы: `+` `-` `*` `/` `//` `%` `**`

`x +` `x -` `x *` `x /` `x //` `x %` `x **`

деление без остатка остаток

`(1+5.3)*2→12.6`

`abs(-3.2)→3.2`

`round(3.57,1)→3.6`

польза в радианах

`from math import sin, pi...`

`sin(pi/4)→0.707...`

`cos(2*pi/3)→-0.4999...`

`acos(0.5)→1.0471...`

`sqrt(81)→9.0` ✓

`log(e**2)→2.0` и т.д. (см. docs)

Блоки инструкций

родительская инструкция:

→ блок инструкций 1...

родительская инструкция:

→ блок инструкций 2...

след. инструкция после блока 1

Шпаргалка Python №2

Цикл с условием

блок инструкций выполняется до тех пор, пока условие истинно

while логическое выражение:

→ блок инструкций

```

s = 0
i = 1
while i <= 100:
    s = s + i**2
    i = i + 1
print("sum:", s)

```

условие с логикой бы одним изменяющимся значением (здесь i)

выражения вычисляются пока i ≤ 100

изменяет переменную цикла

вычисленный результат цикла

остерегайтесь бесконечных циклов!

Управление циклом

break немедленный выход

continue следующая итерация

$s = \sum_{i=1}^{100} i^2$

Цикл перебора

блок инструкций выполняется для всех элементов контейнера или итератора

for переменная in последовательность:

→ блок инструкций

Проход по элементам последовательности

```

s = "Some text"
cnt = 0
for c in s:
    if c == "e":
        cnt = cnt + 1
print("found", cnt, "e")

```

инициализация перед циклом

переменная цикла, значение управляется циклом for

Посчитать число букв e в строке

цикл по dict/set = цикл по последовательности ключей

используйте срезы для прохода по подпоследовательностям

Проход по индексам последовательности

- можно присваивать элемент по индексу
- доступ к соседним элементам

```

lst = [11, 18, 9, 12, 23, 4, 17]
lost = []
for idx in range(len(lst)):
    val = lst[idx]
    if val > 15:
        lost.append(val)
        lst[idx] = 15
print("modif:", lst, "-lost:", lost)

```

Ограничить значения больше 15, запомнить потерянные значения

Пройти одновременно по индексам и значениям:

```
for idx, val in enumerate(lst):
```

Печать / Ввод

```
print("v=", 3, "cm :", x, " ", y+4)
```

элементы для отображения: литералы, переменные, выражения

настройки print:

- sep=" " (разделитель аргументов, по умолч. пробел)
- end="\n" (конец печати, по умолч. перевод строки)
- file=f (печать в файл, по умолч. стандартный вывод)

s = input("Instructions: ")

input всегда возвращает строку, преобразуйте её к нужному типу сами (см. «Преобразования» на другой стороне).

Операции с контейнерами

len(c) → количество элементов

min(c) **max(c)** **sum(c)** Прим.: для словарей и списков эти операции работают с ключами.

sorted(c) → отсортированная копия

val in c → boolean, membership operator (absence not in)

enumerate(c) → итератор по парам (индекс, значение)

Только для последовательностей (lists, tuples, strings):

reversed(c) → reverse iterator

c.index(val) → позиция

c*5 → повторить **c+c2** → соединить

c.count(val) → подсчёт вхождений

изменяют первоначальный список

Операции со списками

lst.append(item) добавляет элемент в конец

lst.extend(seq) добавляет последовательность в конец

lst.insert(idx, val) вставить значение по индексу

lst.remove(val) удалить первое вхождение val

lst.pop(idx) удалить значение по индексу и вернуть его

lst.sort() **lst.reverse()** сортировать/обратить список по месту

Генераторы последовательностей int

часто используются по умолчанию 0 не включается

range([start, stop[, step]])

range(5) → 0 1 2 3 4

range(3, 8) → 3 4 5 6 7

range(2, 12, 3) → 2 5 8 11

range возвращает «генератор», чтобы увидеть значения, преобразуйте его в последовательность, например:

```
print(list(range(4)))
```

Операции со словарями

d[key]=value **d.clear()**

d[key]→value **del d[key]**

d.update(d2) Обновить/добавить пары

d.keys() пары

d.values() просмотр ключей, значений и пар

d.items()

d.pop(key)

Операции с множествами

Операторы:

- | → объединение (вертикальная черта)
- & → пересечение
- → разность/симметричная разн.
- < <= > >= → отношения включения

s.update(s2) **s.add(key)** **s.remove(key)** **s.discard(key)**

Определение функций

имя функции (идентификатор)

именованные параметры

```
def fctname(p_x, p_y, p_z):
    """documentation"""
    # инструкции, вычисление результата
    return res
```

→ инструкции, вычисление результата

→ результат вызова, если нет возврата значения, по умолчанию вернёт None

параметры и весь этот блок существуют только во время вызова функции («черная коробка»)

Вызов функций

```
r = fctname(3, i+2, 2*i)
```

один аргумент каждому параметру

получить результат (если нужен)

Операции со словарями

d[key]=value **d.clear()**

d[key]→value **del d[key]**

d.update(d2) Обновить/добавить пары

d.keys() пары

d.values() просмотр ключей, значений и пар

d.items()

d.pop(key)

Операции с множествами

Операторы:

- | → объединение (вертикальная черта)
- & → пересечение
- → разность/симметричная разн.
- < <= > >= → отношения включения

s.update(s2) **s.add(key)** **s.remove(key)** **s.discard(key)**

Файлы

Создание и считывание файлов с диска

```
f = open("fil.txt", "w", encoding="utf8")
```

файловая переменная для операций

имя файла на диске (+путь...)

режим работы

- 'r' read
- 'w' write
- 'a' append...

кодировка символов в текстовых файлах: utf8, ascii, cp1251 ...

см. функции в модулях **os** и **os.path**

запись

f.write("hello")

текстовый файл → чтение/запись только строк, преобразуйте требуемые типы

пустая строка при конце файла

s = f.read(4) прочитать следующую строку

s = f.readline() прочитать следующую строку

если количество символов не указано, прочитает весь файл

f.close() не забывайте закрывать после использования

Автоматическое закрытие: **with open(...) as f:**

очень часто: цикл по строкам (каждая до '\n') текстового файла

```
for line in f:
```

→ блок кода для обработки строки

Форматирование строк

форматные директивы

значения для форматирования

```
"model {} {} {}".format(x, y, r)
```

{ селектор: формат! преобразование }

Селекторы:

- 2
- x
- 0, nom
- 4[key]
- 0[2]

Примеры:

- "{:+2.3f}".format(45.7273) → '+45.727'
- "{:>10s}".format(8, "toto") → 'toto'
- "{:r}".format('I'm') → 'I\\'m'

Формат:

- автолетные выровненные знак, минимизация, точность, максимизация

< > ^ - пробел 0 в начале для заполнения 0

целые: b - бинарный, c - символ, d - десятичный (по умолч.), e - E-ригн, x или X - 16-ригн

float: e or E - экспоненциальная запись, f or F - фиксир. точка, g or G - наиболее подходящая из e или F, % - перевод долей в %

строки: s ...

Преобразование: s (читаемый текст) или r (в виде литерала)

#метод простой Вставки (insert)

def insert(arr, dim):

mcount = 0 # счетчик перестановок

ccount = 0 # счетчик сравнений

for i in range(1, dim): # Основной цикл со 2-го элемента право

temp = arr[i] # Запомним элемент для сравнения

j = i - 1

while j >= 0: # Ищем влево ближайший меньший

ccount += 1 # Считаем операции сравнения

if arr[j] > temp:

mcount += 1 # Считаем операции перестановки

arr[j+1] = arr[j] # Сдвигаем элемент влево,

а на его место ставим наименьший

arr[j] = temp

j -= 1

return ccount, mcount

```
# -*- coding: utf-8 -*-
# *** Spyder Python Console History Log ***

# ---(Sun Jul 29 17:49:50 2018)-

import numpy as np
import codecs
from prettytable import PrettyTable

import sort_functions

DIM = 40
bubble_arr = []
insert_arr = []
select_arr = []

CTotal = [0, 0, 0]
MTotal = [0, 0, 0]

for i in range(1, DIM+1):
    select_arr.append(i)
    bubble_arr.append(i)
    insert_arr.append(i)

myfile = codecs.open("sort_methods.txt", "w", "utf-8") # Все результаты запишем в файл

print("\nУПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ: Исходный массив")
print(select_arr)

#метод пузырька Select
count = [0, 0]
count = sort_functions.select(select_arr, DIM)
print("УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ: Результирующий массив")
print(select_arr)
CTotal[0] = count[0]
MTotal[0] = count[1]

#метод вставки Insert
count = [0, 0]
count = sort_functions.insert(insert_arr, DIM)
CTotal[1] = count[0]
MTotal[1] = count[1]

#метод пузырька Bubble
count = [0, 0]
count = sort_functions.bubble(bubble_arr, DIM)
```



```

CTotal[2] = count[0]
MTotal[2] = count[1]
print("УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ:")
print("Размер массива: ", DIM)
table = PrettyTable(["Эффективность", "Выбор", "Вставка", "Обмен"])
table.add_row(["Сравнений:", CTotal[0], CTotal[1], CTotal[2]])
table.add_row(["Перестановок:", MTotal[0], MTotal[1], MTotal[2]])
print(table)

wr_str = "УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ:\n"
myfile.write(wr_str)
wr_str = "Размер массива: " + str(DIM) + "\n"
wr_str = wr_str
myfile.write(wr_str)
wr_str = str(table)
wr_str = wr_str
myfile.write(wr_str)

select_arr.clear()
bubble_arr.clear()
insert_arr.clear()

for i in range(DIM, 0, -1):
    select_arr.append(i)
    bubble_arr.append(i)
    insert_arr.append(i)

print("\nОБРАТНО УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ: Исходный массив")
print(select_arr)

#метод пузырька Select
count = [0, 0]
count = sort_functions.select(select_arr, DIM)
print("ОБРАТНО УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ: Результирующий массив")
print(select_arr)
CTotal[0] = count[0]
MTotal[0] = count[1]

#метод вставки Insert
count = [0, 0]
count = sort_functions.insert(insert_arr, DIM)
CTotal[1] = count[0]
MTotal[1] = count[1]

#метод пузырька Bubble
count = [0, 0]

```



```

count = sort_functions.bubble(bubble_arr, DIM)
CTotal[2] = count[0]
MTotal[2] = count[1]
print("Размер массива: ", DIM)

table = PrettyTable(["Эффективность", "Выбор", "Вставка", "Обмен"])
table.add_row(["Сравнений:", CTotal[0], CTotal[1], CTotal[2]])
table.add_row(["Перестановок:", MTotal[0], MTotal[1], MTotal[2]])
print(table)

wr_str = "\nОБРАТНО УПОРЯДОЧЕННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ:\n"
myfile.write(wr_str)
wr_str = "Размер массива: " + str(DIM) + "\n"
myfile.write(wr_str)
wr_str = str(table)
myfile.write(wr_str)

# СЛУЧАЙНАЯ РЕАЛИЗАЦИЯ
NUM = 500
CTotal.clear()
MTotal.clear()

CTotal = [0, 0, 0]
MTotal = [0, 0, 0]

# Организация и проведение численных экспериментов количеством NUM
for n in range(0, NUM):

    select_arr.clear()
    bubble_arr.clear()
    insert_arr.clear()

    select_arr = [np.random.randint(0, 100) for i in range(DIM)]
    for i in range(0, DIM):
        bubble_arr.append(select_arr[i])
        insert_arr.append(select_arr[i])

    #метод пузырька Select
    count = [0, 0]
    count = sort_functions.select(select_arr, DIM)
    CTotal[0] += count[0]
    MTotal[0] += count[1]

    #метод вставки Insert
    count = [0, 0]
    count = sort_functions.insert(insert_arr, DIM)
    CTotal[1] += count[0]
    MTotal[1] += count[1]

```

```

#метод пузырька Bubble
count = [0, 0]
count = sort_functions.bubble(bubble_arr, DIM)
CTotal[2] += count[0]
MTotal[2] += count[1]

print("\nСЛУЧАЙНАЯ РЕАЛИЗАЦИЯ:")
print("Проведено экспериментов: ", NUM)
print("Размер массива: ", DIM)
table = PrettyTable(["Эффективность", "Выбор", "Вставка", "Обмен"])
table.add_row(["Сравнений:", round(CTotal[0]/NUM), round(CTotal[1]/NUM),
round(CTotal[2]/NUM)])
table.add_row(["Перестановок:", round(MTotal[0]/NUM), round(MTotal[1]/NUM),
round(MTotal[2]/NUM)])
print(table)

myfile.write("\nСЛУЧАЙНАЯ РЕАЛИЗАЦИЯ:\n")
wr_str = "Проведено экспериментов: " + str(NUM) + "\n"
myfile.write(wr_str)

wr_str = "Размер массива: " + str(DIM) + "\n"
myfile.write(wr_str)

wr_str = str(table)
myfile.write(wr_str)

myfile.close()
input()

```

Варианты СЛАУ для курсовой работы

Вариант №1

Задача №1

	1.83	1.34	1.74	1.47	1.54	* X1 =	0.71
	1.55	1.13	1.15	1.26	1.44	* X2 =	0.38
	1.04	1.76	1.83	1.25	1.06	* X3 =	0.76
	1.29	1.51	1.31	1.13	1.50	* X4 =	0.38
	1.42	1.95	1.84	1.31	1.55	* X5 =	0.16

Задача №2

	1.05	1.20	1.99	* X1 =	0.68
	1.15	1.02	1.22	* X2 =	0.59
	1.37	1.93	1.16	* X3 =	0.77

Задача №3

	1.61	1.07	1.18	1.03	1.35	* X1 =	0.33
	1.09	1.31	1.78	1.72	1.75	* X2 =	0.84
	1.70	1.78	1.45	1.73	1.93	* X3 =	0.23
	1.71	1.54	1.20	1.14	1.82	* X4 =	0.95
	1.40	1.57	1.68	1.50	1.84	* X5 =	0.72

Задача №4

	1.10	1.20	1.36	1.12	1.41	1.79	* X1 =	0.40
	1.98	1.04	1.10	1.28	1.22	1.94	* X2 =	0.58
	1.07	1.52	1.06	1.65	1.31	1.79	* X3 =	0.45
	1.60	1.24	1.95	1.74	1.32	1.99	* X4 =	0.19
	1.93	1.37	1.16	1.80	1.80	1.64	* X5 =	0.09
	1.15	1.18	1.96	1.61	1.37	1.51	* X6 =	0.11

Задача №5

$$| \quad 1.78 \quad 1.19 \quad | * | X1 | = | \quad 0.26 |$$

$$| \quad 1.68 \quad 1.52 \quad | * | X2 | = | \quad 0.28 |$$

Вариант №2

Задача №1

$$| \quad 1.06 \quad 1.76 \quad 1.47 \quad 1.03 \quad 1.69 \quad | * | X1 | = | \quad 0.10 |$$

$$| \quad 1.74 \quad 1.28 \quad 1.59 \quad 1.49 \quad 1.39 \quad | * | X2 | = | \quad 0.35 |$$

$$| \quad 1.19 \quad 1.11 \quad 1.23 \quad 1.62 \quad 1.39 \quad | * | X3 | = | \quad 0.95 |$$

$$| \quad 1.88 \quad 1.14 \quad 1.34 \quad 1.73 \quad 1.87 \quad | * | X4 | = | \quad 0.81 |$$

$$| \quad 1.68 \quad 1.51 \quad 1.14 \quad 1.52 \quad 1.67 \quad | * | X5 | = | \quad 0.15 |$$

Задача №2

$$| \quad 1.55 \quad 1.33 \quad 1.36 \quad 1.15 \quad 1.90 \quad 1.68 \quad | * | X1 | = | \quad 0.09 |$$

$$| \quad 1.95 \quad 1.29 \quad 1.66 \quad 1.14 \quad 1.10 \quad 1.25 \quad | * | X2 | = | \quad 0.25 |$$

$$| \quad 1.44 \quad 1.21 \quad 1.35 \quad 1.63 \quad 1.80 \quad 1.51 \quad | * | X3 | = | \quad 0.98 |$$

$$| \quad 1.95 \quad 1.48 \quad 1.92 \quad 1.47 \quad 1.20 \quad 1.62 \quad | * | X4 | = | \quad 0.20 |$$

$$| \quad 1.62 \quad 1.54 \quad 1.29 \quad 1.37 \quad 1.73 \quad 1.50 \quad | * | X5 | = | \quad 0.52 |$$

$$| \quad 1.21 \quad 1.21 \quad 1.58 \quad 1.98 \quad 1.03 \quad 1.12 \quad | * | X6 | = | \quad 0.25 |$$

Задача №3

$$| \quad 1.08 \quad 1.55 \quad 1.96 \quad | * | X1 | = | \quad 0.68 |$$

$$| \quad 1.17 \quad 1.90 \quad 1.22 \quad | * | X2 | = | \quad 0.46 |$$

$$| \quad 1.28 \quad 1.33 \quad 1.54 \quad | * | X3 | = | \quad 0.42 |$$

Задача №4

$$| \quad 1.61 \quad 1.34 \quad 1.33 \quad | * | X1 | = | \quad 0.59 |$$

$$| \quad 1.55 \quad 1.35 \quad 1.60 \quad | * | X2 | = | \quad 0.98 |$$

$$| \quad 1.01 \quad 1.64 \quad 1.04 \quad | * | X3 | = | \quad 0.60 |$$

Задача №5

$$| \quad 1.18 \quad 1.16 \quad | * | X1 | = | \quad 0.60 |$$

$$| \quad 1.88 \quad 1.75 \quad | * | X2 | = | \quad 0.99 |$$

Вариант №3

Задача №1

$$| \quad 1.83 \quad 1.94 \quad 1.41 \quad 1.35 \quad | * | X1| = | \quad 0.41|$$

$$| \quad 1.41 \quad 1.60 \quad 1.89 \quad 1.88 \quad | * | X2| = | \quad 0.13|$$

$$| \quad 1.18 \quad 1.19 \quad 1.05 \quad 1.48 \quad | * | X3| = | \quad 0.83|$$

$$| \quad 1.73 \quad 1.67 \quad 1.06 \quad 1.24 \quad | * | X4| = | \quad 0.07|$$

Задача №2

$$| \quad 1.14 \quad 1.09 \quad 1.48 \quad 1.75 \quad 1.94 \quad 1.78 \quad | * | X1| = | \quad 0.17|$$

$$| \quad 1.59 \quad 1.97 \quad 1.99 \quad 1.43 \quad 1.08 \quad 1.23 \quad | * | X2| = | \quad 0.51|$$

$$| \quad 1.87 \quad 1.59 \quad 1.25 \quad 1.51 \quad 1.88 \quad 1.86 \quad | * | X3| = | \quad 0.37|$$

$$| \quad 1.46 \quad 1.16 \quad 1.75 \quad 1.11 \quad 1.19 \quad 1.31 \quad | * | X4| = | \quad 0.96|$$

$$| \quad 1.68 \quad 1.79 \quad 1.21 \quad 1.70 \quad 1.63 \quad 1.66 \quad | * | X5| = | \quad 0.90|$$

$$| \quad 1.36 \quad 1.72 \quad 1.59 \quad 1.10 \quad 1.73 \quad 1.03 \quad | * | X6| = | \quad 0.83|$$

Задача №3

$$| \quad 1.26 \quad 1.86 \quad 1.71 \quad 1.46 \quad | * | X1| = | \quad 0.80|$$

$$| \quad 1.68 \quad 1.24 \quad 1.45 \quad 1.10 \quad | * | X2| = | \quad 0.68|$$

$$| \quad 1.07 \quad 1.60 \quad 1.72 \quad 1.09 \quad | * | X3| = | \quad 0.32|$$

$$| \quad 1.27 \quad 1.23 \quad 1.81 \quad 1.12 \quad | * | X4| = | \quad 0.79|$$

Задача №4

$$| \quad 1.20 \quad 1.89 \quad 1.47 \quad 1.65 \quad 1.05 \quad 1.28 \quad | * | X1| = | \quad 0.24|$$

$$| \quad 1.78 \quad 1.59 \quad 1.77 \quad 1.68 \quad 1.76 \quad 1.63 \quad | * | X2| = | \quad 0.28|$$

$$| \quad 1.39 \quad 1.66 \quad 1.42 \quad 1.67 \quad 1.37 \quad 1.66 \quad | * | X3| = | \quad 0.42|$$

$$| \quad 1.25 \quad 1.82 \quad 1.68 \quad 1.03 \quad 1.78 \quad 1.79 \quad | * | X4| = | \quad 0.12|$$

$$| \quad 1.28 \quad 1.11 \quad 1.74 \quad 1.46 \quad 1.26 \quad 1.17 \quad | * | X5| = | \quad 0.86|$$

$$| \quad 1.38 \quad 1.34 \quad 1.05 \quad 1.83 \quad 1.61 \quad 1.07 \quad | * | X6| = | \quad 0.67|$$

Задача №5

	1.74	1.33	1.98	* X1 =	0.15
	1.16	1.71	1.99	* X2 =	0.17
	1.26	1.05	1.66	* X3 =	0.26

Вариант №4

Задача №1

	1.46	1.95	1.59	* X1 =	0.69
	1.06	1.41	1.14	* X2 =	0.06
	1.94	1.85	1.57	* X3 =	0.69

Задача №2

	1.70	1.51	* X1 =	0.91
	1.17	1.80	* X2 =	0.67

Задача №3

	1.77	1.76	1.61	* X1 =	0.47
	1.09	1.72	1.32	* X2 =	0.55
	1.62	1.80	1.44	* X3 =	0.37

Задача №4

	1.64	1.71	* X1 =	0.66
	1.06	1.56	* X2 =	0.24

Задача №5

	1.66	1.77	1.53	1.08	1.21	* X1 =	0.31
	1.53	1.28	1.10	1.36	1.85	* X2 =	0.57
	1.49	1.59	1.20	1.66	1.30	* X3 =	0.56
	1.37	1.98	1.11	1.22	1.83	* X4 =	0.75
	1.40	1.22	1.83	1.08	1.32	* X5 =	0.94

Вариант №5

Задача №1

$$| \quad 1.81 \quad 1.65 \quad | * | X1 | = | \quad 0.79 |$$

$$| \quad 1.39 \quad 1.27 \quad | * | X2 | = | \quad 0.70 |$$

Задача №2

$$| \quad 1.19 \quad 1.25 \quad 1.87 \quad 1.13 \quad 1.31 \quad | * | X1 | = | \quad 0.95 |$$

$$| \quad 1.64 \quad 1.81 \quad 1.45 \quad 1.71 \quad 1.14 \quad | * | X2 | = | \quad 0.32 |$$

$$| \quad 1.28 \quad 1.98 \quad 1.36 \quad 1.69 \quad 1.88 \quad | * | X3 | = | \quad 0.44 |$$

$$| \quad 1.08 \quad 1.12 \quad 1.66 \quad 1.25 \quad 1.46 \quad | * | X4 | = | \quad 0.23 |$$

$$| \quad 1.03 \quad 1.28 \quad 1.91 \quad 1.44 \quad 1.27 \quad | * | X5 | = | \quad 0.21 |$$

Задача №3

$$| \quad 1.81 \quad 1.32 \quad | * | X1 | = | \quad 0.11 |$$

$$| \quad 1.73 \quad 1.92 \quad | * | X2 | = | \quad 0.04 |$$

Задача №4

$$| \quad 1.99 \quad 1.72 \quad | * | X1 | = | \quad 0.47 |$$

$$| \quad 1.85 \quad 1.69 \quad | * | X2 | = | \quad 0.28 |$$

Задача №5

$$| \quad 1.78 \quad 1.71 \quad | * | X1 | = | \quad 0.26 |$$

$$| \quad 1.25 \quad 1.28 \quad | * | X2 | = | \quad 0.48 |$$

Вариант №6

Задача №1

$$| \quad 1.42 \quad 1.40 \quad 1.88 \quad | * | X1 | = | \quad 0.17 |$$

$$| \quad 1.79 \quad 1.02 \quad 1.49 \quad | * | X2 | = | \quad 0.41 |$$

$$| \quad 1.43 \quad 1.91 \quad 1.65 \quad | * | X3 | = | \quad 0.99 |$$

Задача №2

$$| \quad 1.64 \quad 1.14 \quad 1.54 \quad 1.96 \quad 1.86 \quad 1.92 \quad | * | X1 | = | \quad 0.55 |$$

$$| \quad 1.55 \quad 1.07 \quad 1.54 \quad 1.57 \quad 1.75 \quad 1.09 \quad | * | X2 | = | \quad 0.10 |$$

	1.94	1.50	1.16	1.49	1.79	1.07	* X3 =	0.38
	1.71	1.39	1.33	1.68	1.22	1.44	* X4 =	0.52
	1.72	1.68	1.51	1.78	1.82	1.85	* X5 =	0.02
	1.50	1.50	1.77	1.37	1.98	1.37	* X6 =	0.22

Задача №3

	1.36	1.37	* X1 =	0.51
	1.38	1.90	* X2 =	0.23

Задача №4

	1.26	1.01	* X1 =	0.08
	1.49	1.17	* X2 =	0.70

Задача №5

	1.20	1.74	1.15	1.58	1.69	1.46	* X1 =	0.82
	1.64	1.95	1.11	1.63	1.05	1.62	* X2 =	0.20
	1.14	1.65	1.74	1.77	1.14	1.49	* X3 =	0.68
	1.52	1.79	1.92	1.83	1.52	1.01	* X4 =	0.32
	1.39	1.05	1.08	1.49	1.22	1.92	* X5 =	0.99
	1.13	1.62	1.12	1.21	1.60	1.81	* X6 =	0.52

Вариант №7

Задача №1

	1.69	1.02	1.59	1.11	* X1 =	0.28
	1.08	1.55	1.23	1.65	* X2 =	0.63
	1.50	1.34	1.50	1.13	* X3 =	0.09
	1.86	1.34	1.46	1.66	* X4 =	0.84

Задача №2

	1.11	1.40	1.58	1.04	* X1 =	0.82
	1.61	1.49	1.98	1.48	* X2 =	0.39

	1.45	1.21	1.12	1.04	* X3 =	0.45
	1.38	1.31	1.31	1.86	* X4 =	0.33

Задача №3

	1.52	1.93	1.40	1.54	* X1 =	0.99
	1.61	1.31	1.25	1.83	* X2 =	0.96
	1.81	1.30	1.26	1.61	* X3 =	0.99
	1.73	1.03	1.44	1.12	* X4 =	0.29

Задача №4

	1.32	1.33	1.56	1.86	1.90	1.84	* X1 =	0.15
	1.86	1.57	1.84	1.10	1.91	1.61	* X2 =	0.67
	1.93	1.55	1.49	1.29	1.08	1.89	* X3 =	0.37
	1.08	1.66	1.80	1.20	1.13	1.17	* X4 =	0.08
	1.78	1.79	1.86	1.29	1.72	1.16	* X5 =	0.52
	1.14	1.23	1.05	1.80	1.29	1.39	* X6 =	0.42

Задача №5

	1.15	1.46	1.73	1.70	* X1 =	0.84
	1.89	1.78	1.54	1.49	* X2 =	0.79
	1.09	1.39	1.64	1.02	* X3 =	0.15
	1.62	1.83	1.04	1.18	* X4 =	0.20

Вариант №8

Задача №1

	1.57	1.40	1.71	* X1 =	0.98
	1.71	1.46	1.12	* X2 =	0.54
	1.43	1.98	1.85	* X3 =	0.14

Задача №2

	1.77	1.13	1.17	1.69	1.12	* X1 =	0.02
--	------	------	------	------	------	-----------	------

	1.50	1.19	1.36	1.63	1.33	* X2 =	0.06
	1.01	1.62	1.83	1.84	1.70	* X3 =	0.82
	1.15	1.85	1.48	1.61	1.95	* X4 =	0.08
	1.27	1.26	1.12	1.59	1.39	* X5 =	0.15

Задача №3

	1.62	1.15	1.65	1.62	* X1 =	0.21
	1.78	1.98	1.17	1.81	* X2 =	0.60
	1.20	1.23	1.67	1.14	* X3 =	0.40
	1.97	1.10	1.03	1.04	* X4 =	0.53

Задача №4

	1.61	1.59	1.18	1.85	1.96	* X1 =	0.92
	1.04	1.79	1.04	1.49	1.14	* X2 =	0.58
	1.34	1.68	1.90	1.22	1.30	* X3 =	0.21
	1.31	1.84	1.26	1.39	1.65	* X4 =	0.33
	1.93	1.60	1.65	1.13	1.97	* X5 =	0.86

Задача №5

	1.36	1.90	1.96	1.03	1.72	* X1 =	0.25
	1.94	1.42	1.72	1.79	1.45	* X2 =	0.47
	1.64	1.99	1.29	1.03	1.83	* X3 =	0.85
	1.24	1.74	1.27	1.24	1.46	* X4 =	0.98
	1.78	1.54	1.82	1.88	1.56	* X5 =	0.62

Вариант №9

Задача №1

	1.67	1.97	1.61	1.78	1.31	1.06	* X1 =	0.13
	1.66	1.80	1.91	1.54	1.04	1.57	* X2 =	0.36
	1.31	1.12	1.99	1.63	1.72	1.74	* X3 =	0.39

	1.12	1.97	1.43	1.71	1.60	1.98	* X4 =	0.63
	1.23	1.09	1.73	1.54	1.64	1.03	* X5 =	0.34
	1.40	1.37	1.02	1.71	1.71	1.21	* X6 =	0.64

Задача №2

	1.10	1.34	1.76	1.64	1.21	1.93	* X1 =	0.39
	1.47	1.59	1.04	1.65	1.95	1.32	* X2 =	0.97
	1.33	1.59	1.45	1.16	1.26	1.63	* X3 =	0.95
	1.87	1.09	1.47	1.39	1.38	1.33	* X4 =	0.25
	1.65	1.15	1.37	1.86	1.02	1.34	* X5 =	0.12
	1.64	1.21	1.51	1.21	1.11	1.07	* X6 =	0.84

Задача №3

	1.46	1.26	1.38	* X1 =	0.07
	1.82	1.88	1.78	* X2 =	0.73
	1.69	1.57	1.18	* X3 =	0.71

Задача №4

	1.46	1.90	* X1 =	0.37
	1.58	1.68	* X2 =	0.91

Задача №5

	1.08	1.64	1.43	1.77	1.15	* X1 =	0.66
	1.43	1.92	1.90	1.17	1.23	* X2 =	0.40
	1.85	1.05	1.83	1.29	1.14	* X3 =	0.01
	1.47	1.52	1.01	1.01	1.78	* X4 =	0.10
	1.83	1.58	1.81	1.81	1.01	* X5 =	0.73

Вариант №10

Задача №1

	1.03	1.77	1.08	1.62	1.27	1.69	* X1 =	0.41
--	------	------	------	------	------	------	-----------	------

	1.40	1.35	1.32	1.83	1.65	1.75	* X2 =	0.44
	1.07	1.27	1.17	1.00	1.92	1.82	* X3 =	0.05
	1.27	1.34	1.52	1.80	1.12	1.84	* X4 =	0.31
	1.87	1.49	1.30	1.31	1.46	1.75	* X5 =	0.41
	1.16	1.58	1.52	1.92	1.26	1.97	* X6 =	0.16

Задача №2

	1.30	1.50	* X1 =	0.54
	1.76	1.24	* X2 =	0.66

Задача №3

	1.46	1.34	1.41	1.90	1.33	1.27	* X1 =	0.00
	1.84	1.76	1.28	1.86	1.25	1.34	* X2 =	0.23
	1.42	1.64	1.59	1.49	1.77	1.56	* X3 =	0.95
	1.07	1.56	1.46	1.28	1.03	1.43	* X4 =	0.13
	1.17	1.73	1.95	1.92	1.96	1.46	* X5 =	0.19
	1.76	1.65	1.50	1.35	1.12	1.57	* X6 =	0.72

Задача №4

	1.02	1.09	1.10	1.10	1.96	1.10	* X1 =	0.11
	1.05	1.88	1.28	1.43	1.78	1.85	* X2 =	0.07
	1.90	1.82	1.62	1.52	1.73	1.89	* X3 =	0.21
	1.71	1.14	1.53	1.23	1.42	1.87	* X4 =	0.89
	1.50	1.23	1.92	1.62	1.52	1.49	* X5 =	0.40
	1.41	1.78	1.28	1.30	1.48	1.68	* X6 =	0.41

Задача №5

	1.39	1.80	1.85	1.35	1.73	1.42	* X1 =	0.71
	1.75	1.30	1.41	1.95	1.54	1.94	* X2 =	0.21
	1.29	1.87	1.36	1.74	1.03	1.00	* X3 =	0.50
	1.06	1.17	1.01	1.13	1.91	1.21	* X4 =	0.14

	1.05	1.70	1.95	1.95	1.45	1.99	* X5 =	0.68
	1.25	1.68	1.29	1.43	1.69	1.77	* X6 =	0.56

Вариант №11

Задача №1

	1.34	1.17	* X1 =	0.33
	1.38	1.21	* X2 =	0.41

Задача №2

	1.33	1.89	1.23	1.81	1.30	* X1 =	0.59
	1.30	1.26	1.06	1.07	1.52	* X2 =	0.91
	1.47	1.81	1.36	1.49	1.65	* X3 =	0.50
	1.88	1.90	1.96	1.21	1.90	* X4 =	0.67
	1.65	1.20	1.81	1.80	1.00	* X5 =	0.01

Задача №3

	1.30	1.16	1.45	1.05	1.48	1.12	* X1 =	0.29
	1.89	1.50	1.34	1.94	1.23	1.06	* X2 =	0.03
	1.49	1.80	1.38	1.91	1.36	1.10	* X3 =	0.03
	1.27	1.65	1.92	1.93	1.88	1.85	* X4 =	0.18
	1.40	1.31	1.90	1.76	1.75	1.56	* X5 =	0.67
	1.98	1.37	1.98	1.65	1.24	1.15	* X6 =	0.34

Задача №4

	1.09	1.93	* X1 =	0.58
	1.95	1.59	* X2 =	0.73

Задача №5

	1.21	1.21	* X1 =	0.37
	1.74	1.55	* X2 =	0.58

Вариант №12

Задача №1

$$| \quad 1.27 \quad 1.17 \quad | * | X1 | = | \quad 0.22 |$$

$$| \quad 1.38 \quad 1.48 \quad | * | X2 | = | \quad 0.39 |$$

Задача №2

$$| \quad 1.18 \quad 1.78 \quad 1.12 \quad 1.78 \quad 1.07 \quad | * | X1 | = | \quad 0.59 |$$

$$| \quad 1.94 \quad 1.75 \quad 1.56 \quad 1.44 \quad 1.04 \quad | * | X2 | = | \quad 0.82 |$$

$$| \quad 1.80 \quad 1.53 \quad 1.10 \quad 1.40 \quad 1.41 \quad | * | X3 | = | \quad 0.30 |$$

$$| \quad 1.02 \quad 1.01 \quad 1.16 \quad 1.61 \quad 1.83 \quad | * | X4 | = | \quad 0.22 |$$

$$| \quad 1.62 \quad 1.40 \quad 1.43 \quad 1.76 \quad 1.88 \quad | * | X5 | = | \quad 0.17 |$$

Задача №3

$$| \quad 1.36 \quad 1.59 \quad 1.10 \quad 1.45 \quad | * | X1 | = | \quad 0.83 |$$

$$| \quad 2.00 \quad 1.36 \quad 1.02 \quad 1.32 \quad | * | X2 | = | \quad 0.47 |$$

$$| \quad 1.33 \quad 1.04 \quad 1.94 \quad 1.43 \quad | * | X3 | = | \quad 0.75 |$$

$$| \quad 1.41 \quad 1.17 \quad 1.28 \quad 1.96 \quad | * | X4 | = | \quad 0.22 |$$

Задача №4

$$| \quad 1.95 \quad 1.77 \quad | * | X1 | = | \quad 0.19 |$$

$$| \quad 1.84 \quad 1.56 \quad | * | X2 | = | \quad 0.71 |$$

Задача №5

$$| \quad 1.59 \quad 1.07 \quad 1.77 \quad 1.51 \quad 1.51 \quad | * | X1 | = | \quad 0.54 |$$

$$| \quad 1.56 \quad 1.71 \quad 1.95 \quad 1.75 \quad 1.97 \quad | * | X2 | = | \quad 0.50 |$$

$$| \quad 1.70 \quad 1.20 \quad 1.04 \quad 1.08 \quad 1.24 \quad | * | X3 | = | \quad 0.67 |$$

$$| \quad 1.63 \quad 1.43 \quad 1.01 \quad 1.92 \quad 1.96 \quad | * | X4 | = | \quad 0.23 |$$

$$| \quad 1.03 \quad 1.19 \quad 1.43 \quad 1.22 \quad 1.95 \quad | * | X5 | = | \quad 0.80 |$$

Вариант №13

Задача №1

$$| \quad 1.97 \quad 1.48 \quad | * | X1 | = | \quad 0.64 |$$

$$| \quad 1.97 \quad 1.60 \quad | * | X2| = | \quad 0.10|$$

Задача №2

$$| \quad 1.40 \quad 1.17 \quad 1.45 \quad 1.85 \quad 1.45 \quad | * | X1| = | \quad 0.17|$$

$$| \quad 1.85 \quad 1.04 \quad 1.43 \quad 1.98 \quad 1.83 \quad | * | X2| = | \quad 0.84|$$

$$| \quad 1.40 \quad 1.09 \quad 1.53 \quad 1.27 \quad 1.35 \quad | * | X3| = | \quad 0.95|$$

$$| \quad 1.21 \quad 1.43 \quad 1.78 \quad 1.00 \quad 1.34 \quad | * | X4| = | \quad 0.80|$$

$$| \quad 1.04 \quad 1.94 \quad 1.97 \quad 1.20 \quad 1.90 \quad | * | X5| = | \quad 0.17|$$

Задача №3

$$| \quad 1.05 \quad 1.85 \quad 1.85 \quad 1.57 \quad 1.22 \quad | * | X1| = | \quad 0.22|$$

$$| \quad 1.95 \quad 1.65 \quad 1.39 \quad 1.65 \quad 1.38 \quad | * | X2| = | \quad 0.81|$$

$$| \quad 1.80 \quad 1.83 \quad 1.61 \quad 1.24 \quad 1.84 \quad | * | X3| = | \quad 0.98|$$

$$| \quad 1.01 \quad 1.57 \quad 1.62 \quad 1.47 \quad 1.51 \quad | * | X4| = | \quad 0.54|$$

$$| \quad 1.37 \quad 1.60 \quad 1.26 \quad 1.30 \quad 1.25 \quad | * | X5| = | \quad 0.47|$$

Задача №4

$$| \quad 1.98 \quad 1.16 \quad 1.35 \quad 1.92 \quad | * | X1| = | \quad 0.48|$$

$$| \quad 1.34 \quad 1.16 \quad 1.19 \quad 1.62 \quad | * | X2| = | \quad 0.37|$$

$$| \quad 1.11 \quad 1.40 \quad 1.26 \quad 1.77 \quad | * | X3| = | \quad 0.95|$$

$$| \quad 1.72 \quad 1.41 \quad 1.25 \quad 1.94 \quad | * | X4| = | \quad 0.94|$$

Задача №5

$$| \quad 1.08 \quad 1.21 \quad 1.97 \quad | * | X1| = | \quad 0.26|$$

$$| \quad 1.64 \quad 1.14 \quad 1.78 \quad | * | X2| = | \quad 0.89|$$

$$| \quad 1.54 \quad 1.63 \quad 1.97 \quad | * | X3| = | \quad 0.85|$$

Вариант №14

Задача №1

$$| \quad 1.48 \quad 1.72 \quad 1.75 \quad | * | X1| = | \quad 0.30|$$

$$| \quad 1.45 \quad 1.98 \quad 1.74 \quad | * | X2| = | \quad 0.45|$$

	1.49	1.97	1.48	* X3 =	0.03
--	------	------	------	-----------	------

Задача №2

	1.27	1.82	1.92	1.17	1.07	1.74	* X1 =	0.79
--	------	------	------	------	------	------	-----------	------

	1.63	1.45	1.97	1.72	1.80	1.39	* X2 =	0.19
--	------	------	------	------	------	------	-----------	------

	1.21	1.36	1.64	1.95	1.58	1.65	* X3 =	0.98
--	------	------	------	------	------	------	-----------	------

	1.62	1.28	1.28	1.40	1.13	1.75	* X4 =	0.72
--	------	------	------	------	------	------	-----------	------

	1.22	1.47	1.42	1.54	1.73	1.80	* X5 =	0.39
--	------	------	------	------	------	------	-----------	------

	1.62	1.67	1.03	1.75	1.15	1.66	* X6 =	0.39
--	------	------	------	------	------	------	-----------	------

Задача №3

	1.11	1.17	1.23	1.02	* X1 =	0.75
--	------	------	------	------	-----------	------

	1.28	1.48	1.22	1.84	* X2 =	0.99
--	------	------	------	------	-----------	------

	1.79	1.58	1.81	1.50	* X3 =	0.12
--	------	------	------	------	-----------	------

	1.93	1.12	1.51	1.95	* X4 =	0.95
--	------	------	------	------	-----------	------

Задача №4

	1.86	1.89	1.93	1.16	1.91	1.52	* X1 =	0.05
--	------	------	------	------	------	------	-----------	------

	1.02	1.13	1.33	1.24	1.43	1.30	* X2 =	0.97
--	------	------	------	------	------	------	-----------	------

	1.68	1.80	1.67	1.35	1.44	1.14	* X3 =	0.04
--	------	------	------	------	------	------	-----------	------

	1.26	1.69	1.45	1.91	1.02	1.60	* X4 =	0.90
--	------	------	------	------	------	------	-----------	------

	1.47	1.99	1.89	1.28	1.60	1.55	* X5 =	0.82
--	------	------	------	------	------	------	-----------	------

	1.19	1.37	1.01	1.51	2.00	1.61	* X6 =	0.69
--	------	------	------	------	------	------	-----------	------

Задача №5

	1.60	1.79	1.76	1.99	1.56	1.23	* X1 =	0.62
--	------	------	------	------	------	------	-----------	------

	1.08	1.64	1.06	1.13	1.61	1.52	* X2 =	0.17
--	------	------	------	------	------	------	-----------	------

	1.41	1.86	1.93	1.72	1.50	1.03	* X3 =	0.11
--	------	------	------	------	------	------	-----------	------

	1.81	1.66	1.09	1.02	1.57	1.63	* X4 =	0.98
--	------	------	------	------	------	------	-----------	------

	1.72	1.33	1.65	1.22	1.91	1.32	* X5 =	0.04
--	------	------	------	------	------	------	-----------	------

	1.89	1.91	1.88	1.27	1.46	1.50	* X6 =	0.91
--	------	------	------	------	------	------	-----------	------

Вариант №15

Задача №1

$$| \quad 1.70 \quad 1.08 \quad | * | X1 | = | \quad 0.75 |$$

$$| \quad 1.19 \quad 1.72 \quad | * | X2 | = | \quad 0.43 |$$

Задача №2

$$| \quad 1.38 \quad 1.51 \quad 1.96 \quad | * | X1 | = | \quad 0.26 |$$

$$| \quad 1.18 \quad 1.02 \quad 1.44 \quad | * | X2 | = | \quad 0.94 |$$

$$| \quad 1.76 \quad 1.05 \quad 1.57 \quad | * | X3 | = | \quad 0.15 |$$

Задача №3

$$| \quad 1.63 \quad 1.82 \quad | * | X1 | = | \quad 0.03 |$$

$$| \quad 1.91 \quad 1.24 \quad | * | X2 | = | \quad 0.06 |$$

Задача №4

$$| \quad 1.92 \quad 1.03 \quad 1.90 \quad 1.62 \quad | * | X1 | = | \quad 0.03 |$$

$$| \quad 1.95 \quad 1.06 \quad 1.06 \quad 1.70 \quad | * | X2 | = | \quad 0.51 |$$

$$| \quad 1.84 \quad 1.78 \quad 1.37 \quad 1.33 \quad | * | X3 | = | \quad 0.01 |$$

$$| \quad 1.90 \quad 1.67 \quad 1.95 \quad 1.40 \quad | * | X4 | = | \quad 0.82 |$$

Задача №5

$$| \quad 1.35 \quad 1.96 \quad 1.28 \quad 1.16 \quad 1.90 \quad | * | X1 | = | \quad 0.84 |$$

$$| \quad 1.30 \quad 1.80 \quad 1.73 \quad 1.87 \quad 1.33 \quad | * | X2 | = | \quad 0.62 |$$

$$| \quad 1.08 \quad 1.74 \quad 1.24 \quad 1.02 \quad 1.24 \quad | * | X3 | = | \quad 0.74 |$$

$$| \quad 1.34 \quad 1.29 \quad 1.42 \quad 1.39 \quad 1.92 \quad | * | X4 | = | \quad 0.80 |$$

$$| \quad 1.21 \quad 1.70 \quad 1.76 \quad 1.35 \quad 1.32 \quad | * | X5 | = | \quad 0.44 |$$

Вариант №16

Задача №1

$$| \quad 1.73 \quad 1.96 \quad 1.58 \quad 1.19 \quad | * | X1 | = | \quad 0.68 |$$

$$| \quad 1.71 \quad 1.56 \quad 1.68 \quad 1.21 \quad | * | X2 | = | \quad 0.83 |$$

$$| \quad 1.30 \quad 1.50 \quad 1.94 \quad 1.28 \quad | * | X3 | = | \quad 0.18 |$$

$$| \quad 1.90 \quad 1.48 \quad 1.28 \quad 1.31 \quad | * | X4 | = | \quad 0.43 |$$

Задача №2

$$| \quad 1.03 \quad 1.41 \quad 1.23 \quad | * | X1 | = | \quad 0.27 |$$

$$| \quad 1.93 \quad 1.79 \quad 1.07 \quad | * | X2 | = | \quad 0.12 |$$

$$| \quad 1.00 \quad 1.99 \quad 1.04 \quad | * | X3 | = | \quad 0.57 |$$

Задача №3

$$| \quad 1.40 \quad 1.06 \quad | * | X1 | = | \quad 0.20 |$$

$$| \quad 1.25 \quad 1.22 \quad | * | X2 | = | \quad 0.60 |$$

Задача №4

$$| \quad 1.52 \quad 1.71 \quad | * | X1 | = | \quad 0.67 |$$

$$| \quad 1.02 \quad 1.81 \quad | * | X2 | = | \quad 0.81 |$$

Задача №5

$$| \quad 1.68 \quad 1.98 \quad 1.19 \quad 1.88 \quad | * | X1 | = | \quad 0.69 |$$

$$| \quad 1.99 \quad 1.08 \quad 1.60 \quad 1.85 \quad | * | X2 | = | \quad 0.37 |$$

$$| \quad 1.59 \quad 1.23 \quad 1.72 \quad 1.68 \quad | * | X3 | = | \quad 0.56 |$$

$$| \quad 1.02 \quad 1.66 \quad 1.08 \quad 1.53 \quad | * | X4 | = | \quad 0.78 |$$

Вариант №17

Задача №1

$$| \quad 1.10 \quad 1.90 \quad 1.00 \quad 1.58 \quad 1.70 \quad | * | X1 | = | \quad 0.93 |$$

$$| \quad 1.23 \quad 1.83 \quad 1.48 \quad 1.56 \quad 1.99 \quad | * | X2 | = | \quad 0.57 |$$

$$| \quad 1.16 \quad 1.71 \quad 1.16 \quad 1.68 \quad 1.18 \quad | * | X3 | = | \quad 0.89 |$$

$$| \quad 1.64 \quad 1.91 \quad 1.40 \quad 1.47 \quad 1.28 \quad | * | X4 | = | \quad 0.17 |$$

$$| \quad 1.24 \quad 1.25 \quad 1.45 \quad 1.69 \quad 1.20 \quad | * | X5 | = | \quad 0.80 |$$

Задача №2

$$| \quad 1.11 \quad 1.89 \quad 1.28 \quad | * | X1 | = | \quad 0.90 |$$

$$| \quad 1.99 \quad 1.98 \quad 1.60 \quad | * | X2 | = | \quad 0.57 |$$

$$| \quad 1.72 \quad 1.57 \quad 1.37 \quad | * | X3 | = | \quad 0.45 |$$

Задача №3

$$| \quad 1.73 \quad 1.17 \quad | * | X1 | = | \quad 0.78 |$$

$$| \quad 1.86 \quad 1.07 \quad | * | X2 | = | \quad 0.25 |$$

Задача №4

$$| \quad 1.91 \quad 1.40 \quad | * | X1 | = | \quad 0.53 |$$

$$| \quad 1.92 \quad 1.74 \quad | * | X2 | = | \quad 0.76 |$$

Задача №5

$$| \quad 1.65 \quad 1.79 \quad 1.37 \quad 1.19 \quad 1.83 \quad | * | X1 | = | \quad 0.68 |$$

$$| \quad 1.05 \quad 1.61 \quad 1.35 \quad 1.67 \quad 1.56 \quad | * | X2 | = | \quad 0.78 |$$

$$| \quad 2.00 \quad 1.28 \quad 1.22 \quad 1.79 \quad 1.62 \quad | * | X3 | = | \quad 0.30 |$$

$$| \quad 1.94 \quad 1.79 \quad 1.99 \quad 1.80 \quad 1.57 \quad | * | X4 | = | \quad 0.37 |$$

$$| \quad 1.53 \quad 1.74 \quad 1.32 \quad 1.01 \quad 1.75 \quad | * | X5 | = | \quad 0.19 |$$

Вариант №18

Задача №1

$$| \quad 1.07 \quad 1.79 \quad 1.95 \quad | * | X1 | = | \quad 0.96 |$$

$$| \quad 1.52 \quad 1.97 \quad 1.37 \quad | * | X2 | = | \quad 0.01 |$$

$$| \quad 1.43 \quad 1.26 \quad 1.02 \quad | * | X3 | = | \quad 0.80 |$$

Задача №2

$$| \quad 1.01 \quad 1.15 \quad | * | X1 | = | \quad 0.37 |$$

$$| \quad 1.20 \quad 1.87 \quad | * | X2 | = | \quad 0.82 |$$

Задача №3

$$| \quad 1.51 \quad 1.68 \quad 1.48 \quad 1.42 \quad | * | X1 | = | \quad 0.01 |$$

$$| \quad 1.01 \quad 1.75 \quad 1.49 \quad 1.04 \quad | * | X2 | = | \quad 0.39 |$$

$$| \quad 1.85 \quad 1.50 \quad 1.71 \quad 1.13 \quad | * | X3 | = | \quad 0.74 |$$

$$| \quad 1.33 \quad 1.24 \quad 1.62 \quad 1.18 \quad | * | X4 | = | \quad 0.56 |$$

Задача №4

$$| \quad 1.92 \quad 1.61 \quad | * | X1 | = | \quad 0.14 |$$

$$| \quad 1.29 \quad 1.90 \quad | * | X2 | = | \quad 0.69 |$$

Задача №5

$$| \quad 1.09 \quad 1.86 \quad 1.50 \quad 1.22 \quad 1.73 \quad | * | X1 | = | \quad 0.71 |$$

$$| \quad 1.93 \quad 1.20 \quad 1.86 \quad 1.09 \quad 1.16 \quad | * | X2 | = | \quad 0.92 |$$

$$| \quad 1.00 \quad 1.70 \quad 1.14 \quad 1.43 \quad 1.46 \quad | * | X3 | = | \quad 0.95 |$$

$$| \quad 1.92 \quad 1.68 \quad 1.48 \quad 1.80 \quad 1.54 \quad | * | X4 | = | \quad 0.24 |$$

$$| \quad 1.61 \quad 1.39 \quad 1.76 \quad 1.65 \quad 1.13 \quad | * | X5 | = | \quad 0.38 |$$

Вариант №19

Задача №1

$$| \quad 1.26 \quad 1.99 \quad 1.40 \quad 1.71 \quad | * | X1 | = | \quad 0.07 |$$

$$| \quad 1.62 \quad 1.02 \quad 1.08 \quad 1.61 \quad | * | X2 | = | \quad 0.61 |$$

$$| \quad 1.05 \quad 1.13 \quad 1.91 \quad 1.51 \quad | * | X3 | = | \quad 0.76 |$$

$$| \quad 1.41 \quad 1.69 \quad 1.47 \quad 1.58 \quad | * | X4 | = | \quad 0.56 |$$

Задача №2

$$| \quad 1.35 \quad 1.79 \quad 1.32 \quad 1.91 \quad 1.82 \quad 1.34 \quad | * | X1 | = | \quad 0.06 |$$

$$| \quad 1.70 \quad 1.28 \quad 1.57 \quad 1.85 \quad 1.54 \quad 1.17 \quad | * | X2 | = | \quad 0.89 |$$

$$| \quad 1.60 \quad 1.82 \quad 1.36 \quad 1.29 \quad 1.10 \quad 1.48 \quad | * | X3 | = | \quad 0.17 |$$

$$| \quad 1.43 \quad 1.16 \quad 1.60 \quad 1.22 \quad 1.97 \quad 1.96 \quad | * | X4 | = | \quad 0.56 |$$

$$| \quad 1.92 \quad 1.92 \quad 1.50 \quad 1.56 \quad 1.74 \quad 1.85 \quad | * | X5 | = | \quad 0.90 |$$

$$| \quad 1.11 \quad 1.66 \quad 1.42 \quad 1.07 \quad 1.48 \quad 1.25 \quad | * | X6 | = | \quad 0.16 |$$

Задача №3

$$| \quad 1.48 \quad 1.54 \quad 1.41 \quad 1.34 \quad 1.26 \quad | * | X1 | = | \quad 0.43 |$$

$$| \quad 1.09 \quad 1.57 \quad 1.56 \quad 1.36 \quad 1.79 \quad | * | X2 | = | \quad 0.35 |$$

	1.43	1.70	1.27	1.41	1.64	* X3 =	0.58
	1.22	1.45	1.12	1.25	1.72	* X4 =	0.11
	1.76	1.41	1.11	1.99	1.11	* X5 =	0.65

Задача №4

	1.90	1.03	1.93	1.65	* X1 =	0.44
	1.88	1.50	1.49	1.32	* X2 =	0.59
	1.31	1.17	1.40	1.20	* X3 =	0.73
	1.09	1.77	1.31	1.58	* X4 =	0.91

Задача №5

	1.05	1.32	1.53	1.48	* X1 =	0.87
	1.54	1.93	1.44	1.59	* X2 =	0.87
	1.39	1.21	1.47	1.25	* X3 =	0.59
	1.49	1.78	1.60	1.67	* X4 =	0.03

Вариант №20

Задача №1

	1.80	1.77	1.48	1.39	1.24	1.30	* X1 =	0.75
	1.80	1.67	1.97	1.34	1.12	1.32	* X2 =	0.51
	1.91	1.54	1.49	1.61	1.60	1.43	* X3 =	0.66
	1.49	1.86	1.36	1.99	1.53	1.85	* X4 =	0.58
	1.64	1.67	1.74	1.13	1.84	1.52	* X5 =	0.88
	1.27	1.12	1.80	1.28	1.04	1.67	* X6 =	0.02

Задача №2

	1.82	1.78	1.51	1.38	1.03	1.28	* X1 =	0.75
	1.30	1.27	1.61	1.19	1.85	1.63	* X2 =	0.75
	1.59	1.30	1.62	1.65	1.69	1.81	* X3 =	0.62
	1.97	1.12	1.58	1.97	1.65	1.66	* X4 =	0.50

	1.07	1.08	1.22	1.50	1.57	1.85	* X5 =	0.29
	1.34	1.23	1.92	1.82	1.57	1.54	* X6 =	0.76

Задача №3

	1.68	1.24	1.80	1.62	* X1 =	0.67
	1.06	1.70	1.83	1.10	* X2 =	0.08
	1.40	1.13	1.24	1.02	* X3 =	0.88
	1.58	1.01	1.36	1.93	* X4 =	0.40

Задача №4

	1.13	1.58	1.12	* X1 =	0.06
	1.88	1.30	1.31	* X2 =	0.69
	1.61	1.75	1.67	* X3 =	0.33

Задача №5

	1.04	1.25	* X1 =	0.44
	1.09	1.17	* X2 =	0.60

Вариант №21

Задача №1

	1.81	1.76	* X1 =	0.56
	1.09	1.08	* X2 =	0.59

Задача №2

	1.78	1.20	1.33	1.50	* X1 =	0.53
	1.57	1.04	1.57	1.88	* X2 =	0.13
	1.98	1.49	1.56	1.96	* X3 =	0.27
	1.79	1.78	1.66	1.55	* X4 =	0.47

Задача №3

	1.54	1.71	1.09	1.48	* X1 =	0.83
	1.73	1.11	1.09	1.13	* X2 =	0.26

$$| \quad 1.19 \quad 1.07 \quad 1.62 \quad 1.44 \quad | * | X3 | = | \quad 0.14 |$$

$$| \quad 1.70 \quad 1.48 \quad 1.34 \quad 1.30 \quad | * | X4 | = | \quad 0.13 |$$

Задача №4

$$| \quad 1.62 \quad 1.10 \quad | * | X1 | = | \quad 0.37 |$$

$$| \quad 1.73 \quad 1.36 \quad | * | X2 | = | \quad 0.92 |$$

Задача №5

$$| \quad 1.25 \quad 1.60 \quad 1.68 \quad 1.77 \quad 1.16 \quad 1.49 \quad | * | X1 | = | \quad 0.95 |$$

$$| \quad 1.42 \quad 1.72 \quad 1.75 \quad 1.93 \quad 1.66 \quad 1.20 \quad | * | X2 | = | \quad 0.43 |$$

$$| \quad 1.79 \quad 1.31 \quad 1.11 \quad 1.31 \quad 1.32 \quad 1.83 \quad | * | X3 | = | \quad 0.75 |$$

$$| \quad 1.35 \quad 1.17 \quad 1.43 \quad 2.00 \quad 1.21 \quad 1.00 \quad | * | X4 | = | \quad 0.41 |$$

$$| \quad 1.32 \quad 1.29 \quad 1.04 \quad 1.99 \quad 1.66 \quad 1.93 \quad | * | X5 | = | \quad 0.87 |$$

$$| \quad 1.53 \quad 1.72 \quad 1.71 \quad 1.29 \quad 1.38 \quad 1.88 \quad | * | X6 | = | \quad 0.70 |$$

Вариант №22

Задача №1

$$| \quad 1.63 \quad 1.48 \quad | * | X1 | = | \quad 0.98 |$$

$$| \quad 1.04 \quad 1.29 \quad | * | X2 | = | \quad 0.55 |$$

Задача №2

$$| \quad 1.57 \quad 1.27 \quad 1.44 \quad 1.07 \quad | * | X1 | = | \quad 0.48 |$$

$$| \quad 1.08 \quad 1.59 \quad 1.77 \quad 1.74 \quad | * | X2 | = | \quad 0.11 |$$

$$| \quad 1.03 \quad 1.31 \quad 1.06 \quad 1.61 \quad | * | X3 | = | \quad 0.04 |$$

$$| \quad 1.61 \quad 1.43 \quad 1.68 \quad 1.32 \quad | * | X4 | = | \quad 0.57 |$$

Задача №3

$$| \quad 1.35 \quad 1.36 \quad 1.15 \quad 1.84 \quad 1.67 \quad 1.23 \quad | * | X1 | = | \quad 0.13 |$$

$$| \quad 1.85 \quad 1.54 \quad 1.30 \quad 1.57 \quad 1.47 \quad 1.32 \quad | * | X2 | = | \quad 0.19 |$$

$$| \quad 1.23 \quad 1.89 \quad 1.83 \quad 1.74 \quad 1.89 \quad 1.19 \quad | * | X3 | = | \quad 0.18 |$$

$$| \quad 1.53 \quad 1.66 \quad 1.48 \quad 1.36 \quad 1.77 \quad 1.45 \quad | * | X4 | = | \quad 0.80 |$$

	1.96	1.97	1.41	1.47	1.84	1.11	* X5 =	0.90
	1.39	1.33	1.95	1.92	1.74	1.20	* X6 =	0.18

Задача №4

	1.93	1.52	1.92	1.30	1.06	1.52	* X1 =	0.60
	1.41	1.53	1.19	1.69	1.18	1.71	* X2 =	0.34
	1.16	1.06	1.08	1.39	1.25	1.41	* X3 =	0.84
	1.16	1.49	1.91	1.30	1.97	1.30	* X4 =	0.65
	1.18	1.68	1.14	1.66	1.55	1.90	* X5 =	0.41
	1.43	1.86	1.48	1.39	1.42	1.12	* X6 =	0.22

Задача №5

	2.00	1.66	1.40	1.72	* X1 =	0.90
	1.88	1.19	1.10	1.56	* X2 =	0.93
	1.12	1.85	1.57	1.30	* X3 =	0.31
	1.69	1.83	1.29	1.92	* X4 =	0.31

Вариант №23

Задача №1

	1.26	1.35	1.46	* X1 =	0.64
	1.75	1.64	1.30	* X2 =	0.14
	1.09	1.55	1.34	* X3 =	0.76

Задача №2

	1.13	1.83	1.12	1.02	* X1 =	0.30
	1.70	1.10	1.89	1.03	* X2 =	0.16
	1.27	1.25	1.54	1.61	* X3 =	0.71
	1.27	1.23	1.28	1.40	* X4 =	0.43

Задача №3

	1.50	1.00	1.80	* X1 =	0.46
--	------	------	------	-----------	------

$$| \quad 1.17 \quad 1.80 \quad 1.05 \quad | * | X2 | = | \quad 0.37 |$$

$$| \quad 1.27 \quad 1.53 \quad 1.40 \quad | * | X3 | = | \quad 0.27 |$$

Задача №4

$$| \quad 1.87 \quad 1.51 \quad | * | X1 | = | \quad 0.66 |$$

$$| \quad 1.42 \quad 1.52 \quad | * | X2 | = | \quad 0.20 |$$

Задача №5

$$| \quad 1.48 \quad 1.26 \quad | * | X1 | = | \quad 0.41 |$$

$$| \quad 1.51 \quad 1.70 \quad | * | X2 | = | \quad 0.50 |$$

Вариант №24

Задача №1

$$| \quad 1.98 \quad 1.43 \quad 1.68 \quad | * | X1 | = | \quad 0.24 |$$

$$| \quad 1.69 \quad 1.76 \quad 1.80 \quad | * | X2 | = | \quad 0.48 |$$

$$| \quad 1.79 \quad 1.77 \quad 1.49 \quad | * | X3 | = | \quad 0.90 |$$

Задача №2

$$| \quad 1.97 \quad 1.70 \quad 1.56 \quad 1.00 \quad | * | X1 | = | \quad 0.41 |$$

$$| \quad 1.12 \quad 1.23 \quad 1.89 \quad 1.51 \quad | * | X2 | = | \quad 0.96 |$$

$$| \quad 1.96 \quad 1.11 \quad 1.64 \quad 1.13 \quad | * | X3 | = | \quad 0.37 |$$

$$| \quad 1.67 \quad 1.71 \quad 1.10 \quad 1.41 \quad | * | X4 | = | \quad 0.87 |$$

Задача №3

$$| \quad 1.48 \quad 1.29 \quad 1.02 \quad 1.16 \quad | * | X1 | = | \quad 0.91 |$$

$$| \quad 1.40 \quad 1.92 \quad 1.96 \quad 2.00 \quad | * | X2 | = | \quad 0.24 |$$

$$| \quad 1.01 \quad 1.92 \quad 1.94 \quad 1.27 \quad | * | X3 | = | \quad 0.42 |$$

$$| \quad 1.41 \quad 1.02 \quad 1.27 \quad 1.24 \quad | * | X4 | = | \quad 0.04 |$$

Задача №4

$$| \quad 1.39 \quad 1.69 \quad | * | X1 | = | \quad 0.26 |$$

$$| \quad 1.86 \quad 1.25 \quad | * | X2 | = | \quad 0.86 |$$

Задача №5

$$| \quad 2.00 \quad 1.06 \quad 1.20 \quad | * | X1 | = | \quad 0.57 |$$

$$| \quad 1.98 \quad 1.93 \quad 1.44 \quad | * | X2 | = | \quad 0.22 |$$

$$| \quad 1.28 \quad 1.32 \quad 1.40 \quad | * | X3 | = | \quad 0.57 |$$

Вариант №25

Задача №1

$$| \quad 1.77 \quad 1.63 \quad 1.43 \quad | * | X1 | = | \quad 0.93 |$$

$$| \quad 1.31 \quad 1.62 \quad 1.37 \quad | * | X2 | = | \quad 0.17 |$$

$$| \quad 1.78 \quad 1.24 \quad 1.49 \quad | * | X3 | = | \quad 0.85 |$$

Задача №2

$$| \quad 1.74 \quad 1.61 \quad 1.09 \quad | * | X1 | = | \quad 0.93 |$$

$$| \quad 1.41 \quad 1.42 \quad 1.90 \quad | * | X2 | = | \quad 0.47 |$$

$$| \quad 1.59 \quad 1.95 \quad 1.92 \quad | * | X3 | = | \quad 0.91 |$$

Задача №3

$$| \quad 1.46 \quad 1.45 \quad 1.59 \quad 1.33 \quad 1.57 \quad | * | X1 | = | \quad 0.39 |$$

$$| \quad 1.28 \quad 1.54 \quad 1.10 \quad 1.88 \quad 1.95 \quad | * | X2 | = | \quad 0.84 |$$

$$| \quad 1.57 \quad 1.80 \quad 1.58 \quad 1.98 \quad 1.19 \quad | * | X3 | = | \quad 0.24 |$$

$$| \quad 1.74 \quad 1.32 \quad 1.48 \quad 1.88 \quad 1.40 \quad | * | X4 | = | \quad 0.06 |$$

$$| \quad 1.42 \quad 1.67 \quad 1.88 \quad 1.32 \quad 1.37 \quad | * | X5 | = | \quad 0.80 |$$

Задача №4

$$| \quad 1.94 \quad 1.69 \quad | * | X1 | = | \quad 0.90 |$$

$$| \quad 1.21 \quad 1.44 \quad | * | X2 | = | \quad 0.62 |$$

Задача №5

$$| \quad 1.98 \quad 1.97 \quad 1.79 \quad 1.51 \quad | * | X1 | = | \quad 0.41 |$$

$$| \quad 1.76 \quad 1.19 \quad 1.29 \quad 1.39 \quad | * | X2 | = | \quad 0.32 |$$

$$| \quad 1.91 \quad 1.80 \quad 1.63 \quad 1.44 \quad | * | X3 | = | \quad 0.62 |$$

	1.75	1.03	1.64	1.85	* X4 =	0.46
--	------	------	------	------	-----------	------

Вариант №26

Задача №1

	1.23	1.40	1.39	1.33	1.14	* X1 =	0.06
--	------	------	------	------	------	-----------	------

	1.32	1.04	1.55	1.63	1.91	* X2 =	0.68
--	------	------	------	------	------	-----------	------

	1.13	1.48	1.05	1.05	1.62	* X3 =	0.98
--	------	------	------	------	------	-----------	------

	1.10	1.04	1.17	1.67	1.89	* X4 =	0.09
--	------	------	------	------	------	-----------	------

	1.10	1.50	1.72	1.45	1.86	* X5 =	0.24
--	------	------	------	------	------	-----------	------

Задача №2

	1.97	1.68	1.10	1.83	1.85	1.94	* X1 =	0.16
--	------	------	------	------	------	------	-----------	------

	1.18	1.97	1.87	1.56	1.74	1.03	* X2 =	0.70
--	------	------	------	------	------	------	-----------	------

	1.20	1.86	1.58	1.44	1.58	1.48	* X3 =	0.29
--	------	------	------	------	------	------	-----------	------

	1.86	1.52	1.93	1.23	1.35	1.27	* X4 =	0.91
--	------	------	------	------	------	------	-----------	------

	1.77	1.44	1.66	1.24	1.33	1.06	* X5 =	0.75
--	------	------	------	------	------	------	-----------	------

	1.97	1.04	1.01	1.71	1.39	1.54	* X6 =	0.41
--	------	------	------	------	------	------	-----------	------

Задача №3

	1.33	1.30	1.31	1.65	1.26	* X1 =	0.47
--	------	------	------	------	------	-----------	------

	1.35	1.34	1.10	1.56	1.06	* X2 =	0.23
--	------	------	------	------	------	-----------	------

	1.44	1.71	1.44	1.46	1.30	* X3 =	0.72
--	------	------	------	------	------	-----------	------

	1.23	1.52	1.89	1.69	1.36	* X4 =	0.36
--	------	------	------	------	------	-----------	------

	1.84	1.37	1.64	1.59	1.56	* X5 =	0.78
--	------	------	------	------	------	-----------	------

Задача №4

	1.52	1.11	* X1 =	0.24
--	------	------	-----------	------

	1.26	1.18	* X2 =	0.19
--	------	------	-----------	------

Задача №5

	1.61	1.66	1.26	1.63	1.58	* X1 =	0.66
--	------	------	------	------	------	-----------	------

	1.05	1.25	1.25	1.96	1.63	* X2 =	0.90
	1.15	1.97	1.93	1.43	1.58	* X3 =	0.65
	1.95	1.33	1.79	1.25	1.31	* X4 =	0.63
	1.82	1.82	1.67	1.86	1.84	* X5 =	0.52

Вариант №27

Задача №1

	1.13	1.98	1.18	1.44	1.06	1.06 * X1 =	0.62
	1.38	1.01	1.82	1.51	1.25	1.23 * X2 =	0.95
	1.84	1.80	1.68	1.35	1.22	1.53 * X3 =	0.85
	1.08	1.66	1.68	1.04	1.82	1.73 * X4 =	0.47
	1.84	1.11	1.95	1.87	1.44	1.21 * X5 =	0.56
	1.54	1.09	1.36	1.42	1.98	1.52 * X6 =	0.21

Задача №2

	1.80	1.95	* X1 =	0.90
	1.62	1.66	* X2 =	0.55

Задача №3

	1.73	1.21	1.73	1.02	* X1 =	0.04
	1.51	1.66	1.71	1.60	* X2 =	0.37
	1.08	1.33	1.63	1.15	* X3 =	0.16
	1.56	1.14	1.81	1.41	* X4 =	0.31

Задача №4

	1.96	1.09	1.88	1.92	1.12	1.56 * X1 =	0.89
	1.26	1.54	1.75	1.59	1.40	1.14 * X2 =	0.22
	1.21	1.39	1.13	1.00	1.05	1.80 * X3 =	0.91
	1.60	1.15	1.20	2.00	1.98	1.48 * X4 =	0.85
	1.11	1.36	1.19	1.65	1.20	1.96 * X5 =	0.99

$$| \quad 1.71 \quad 1.32 \quad 1.36 \quad 1.59 \quad 1.14 \quad 1.52 \quad | * | X6| = | \quad 0.43|$$

Задача №5

$$| \quad 1.25 \quad 1.34 \quad 1.78 \quad | * | X1| = | \quad 0.90|$$

$$| \quad 1.21 \quad 1.31 \quad 1.41 \quad | * | X2| = | \quad 0.41|$$

$$| \quad 1.88 \quad 1.38 \quad 1.27 \quad | * | X3| = | \quad 0.64|$$

Вариант №28

Задача №1

$$| \quad 1.17 \quad 1.70 \quad 1.83 \quad | * | X1| = | \quad 0.18|$$

$$| \quad 1.18 \quad 1.41 \quad 1.67 \quad | * | X2| = | \quad 0.71|$$

$$| \quad 1.96 \quad 1.77 \quad 1.83 \quad | * | X3| = | \quad 0.38|$$

Задача №2

$$| \quad 1.85 \quad 1.87 \quad | * | X1| = | \quad 0.11|$$

$$| \quad 1.24 \quad 1.96 \quad | * | X2| = | \quad 0.74|$$

Задача №3

$$| \quad 1.97 \quad 1.02 \quad 1.32 \quad 1.46 \quad | * | X1| = | \quad 0.04|$$

$$| \quad 1.81 \quad 1.89 \quad 1.62 \quad 1.69 \quad | * | X2| = | \quad 0.90|$$

$$| \quad 1.31 \quad 1.07 \quad 1.21 \quad 1.04 \quad | * | X3| = | \quad 0.79|$$

$$| \quad 1.71 \quad 1.84 \quad 1.04 \quad 1.35 \quad | * | X4| = | \quad 0.80|$$

Задача №4

$$| \quad 1.26 \quad 1.87 \quad 1.05 \quad | * | X1| = | \quad 0.90|$$

$$| \quad 1.69 \quad 1.09 \quad 1.53 \quad | * | X2| = | \quad 0.89|$$

$$| \quad 1.59 \quad 1.44 \quad 1.09 \quad | * | X3| = | \quad 0.34|$$

Задача №5

$$| \quad 1.58 \quad 1.51 \quad | * | X1| = | \quad 0.42|$$

$$| \quad 1.85 \quad 1.19 \quad | * | X2| = | \quad 0.61|$$

Вариант №29

Задача №1

$$| \quad 1.18 \quad 1.86 \quad | * | X1 | = | \quad 0.34 |$$

$$| \quad 1.01 \quad 1.75 \quad | * | X2 | = | \quad 0.44 |$$

Задача №2

$$| \quad 1.05 \quad 1.05 \quad 1.20 \quad 1.21 \quad 1.35 \quad 1.26 \quad | * | X1 | = | \quad 0.12 |$$

$$| \quad 1.67 \quad 1.41 \quad 1.11 \quad 1.03 \quad 1.40 \quad 1.93 \quad | * | X2 | = | \quad 0.34 |$$

$$| \quad 1.80 \quad 1.89 \quad 1.44 \quad 1.42 \quad 1.49 \quad 1.02 \quad | * | X3 | = | \quad 0.46 |$$

$$| \quad 1.96 \quad 1.36 \quad 1.45 \quad 1.68 \quad 1.78 \quad 1.01 \quad | * | X4 | = | \quad 0.63 |$$

$$| \quad 1.27 \quad 1.54 \quad 1.06 \quad 1.30 \quad 1.30 \quad 1.93 \quad | * | X5 | = | \quad 0.00 |$$

$$| \quad 1.24 \quad 1.27 \quad 1.51 \quad 1.32 \quad 1.17 \quad 1.93 \quad | * | X6 | = | \quad 0.36 |$$

Задача №3

$$| \quad 1.22 \quad 1.53 \quad | * | X1 | = | \quad 0.30 |$$

$$| \quad 1.98 \quad 1.14 \quad | * | X2 | = | \quad 0.66 |$$

Задача №4

$$| \quad 1.44 \quad 1.48 \quad 1.39 \quad | * | X1 | = | \quad 0.81 |$$

$$| \quad 1.96 \quad 1.60 \quad 1.82 \quad | * | X2 | = | \quad 0.40 |$$

$$| \quad 1.10 \quad 1.88 \quad 1.03 \quad | * | X3 | = | \quad 0.64 |$$

Задача №5

$$| \quad 1.98 \quad 1.93 \quad 1.16 \quad 1.39 \quad 1.14 \quad 1.25 \quad | * | X1 | = | \quad 0.24 |$$

$$| \quad 1.36 \quad 1.48 \quad 1.28 \quad 1.87 \quad 1.87 \quad 1.73 \quad | * | X2 | = | \quad 0.53 |$$

$$| \quad 1.30 \quad 1.87 \quad 1.23 \quad 1.88 \quad 1.63 \quad 1.96 \quad | * | X3 | = | \quad 0.89 |$$

$$| \quad 1.08 \quad 1.36 \quad 1.49 \quad 1.47 \quad 1.70 \quad 1.58 \quad | * | X4 | = | \quad 0.57 |$$

$$| \quad 1.60 \quad 1.15 \quad 1.52 \quad 1.10 \quad 1.12 \quad 1.78 \quad | * | X5 | = | \quad 0.93 |$$

$$| \quad 1.05 \quad 1.71 \quad 1.04 \quad 1.33 \quad 1.31 \quad 1.40 \quad | * | X6 | = | \quad 0.01 |$$

Вариант №30

Задача №1

	1.20	1.47	1.64	1.58	1.25	1.09	* X1 =	0.13
	1.70	1.11	1.83	1.67	1.04	1.57	* X2 =	0.05
	1.04	1.75	1.20	1.43	1.43	1.45	* X3 =	0.16
	1.59	1.83	1.69	1.01	1.17	1.12	* X4 =	1.00
	1.78	1.20	1.57	1.27	1.09	1.96	* X5 =	0.42
	1.85	1.51	1.98	1.82	1.04	1.56	* X6 =	0.97

Задача №2

	1.14	1.26	1.73	1.70	* X1 =	0.70
	1.75	1.41	1.45	1.06	* X2 =	0.09
	1.32	1.01	1.96	1.83	* X3 =	0.74
	1.06	1.44	1.87	1.93	* X4 =	0.08

Задача №3

	1.84	1.91	* X1 =	0.54
	1.13	1.15	* X2 =	0.37

Задача №4

	1.35	1.51	* X1 =	0.24
	1.97	1.50	* X2 =	0.11

Задача №5

	1.37	1.36	1.36	* X1 =	0.34
	1.65	1.11	1.41	* X2 =	0.20
	1.94	1.29	1.06	* X3 =	0.34