

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЁТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.1
дисциплины
«Основы кроссплатформенного программирования»

Выполнил:
Костукайло Кирилл Николаевич
1 курс, группа ИВТ-б-о-21-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Проверил:
Кафедры инфокоммуникаций, старший
преподаватель
Воронкин Р.А.

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

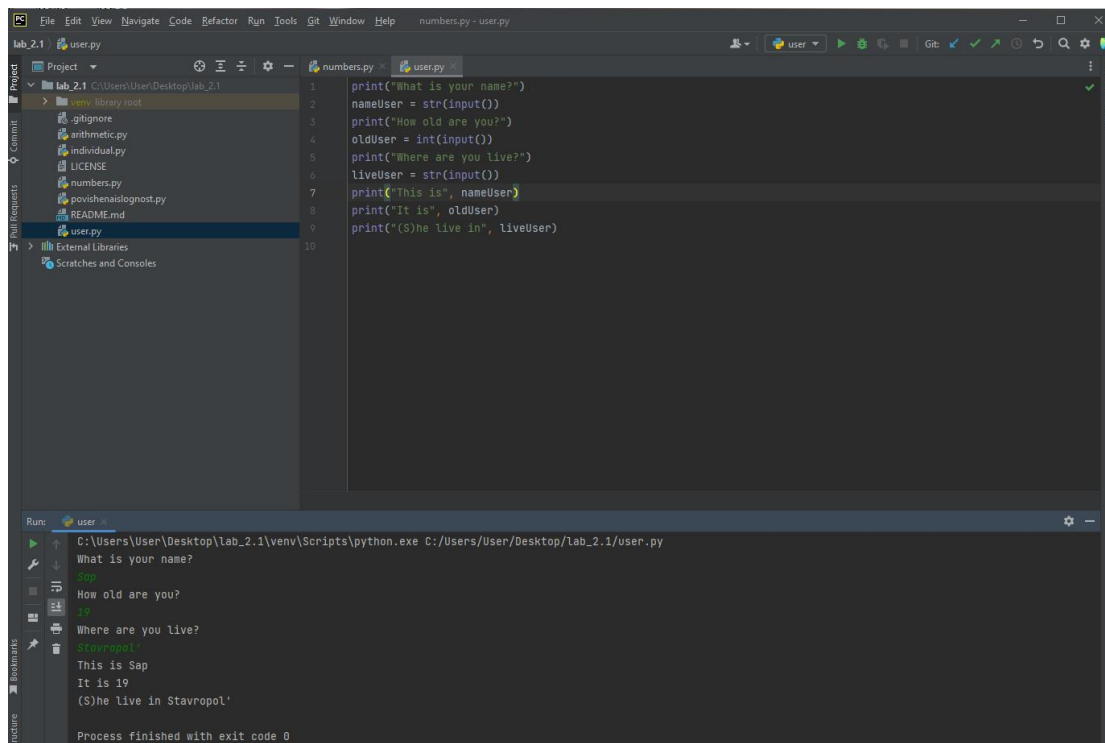
Ставрополь, 2022 г.

Тема: Основы языка Python

Цель работы: исследование процесса установки и базовых возможностей языка Python версии 3.x.

Порядок выполнения работы:

1. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
2. Выполните клонирование созданного репозитория.
3. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
4. Организуйте свой репозиторий в соответствие с моделью ветвления git flow.
5. Создайте проект PyCharm в папке репозитория.
6. Решите следующие задачи с помощью языка программирования Python3 и IDE PyCharm



The screenshot displays the PyCharm IDE interface. The left sidebar shows the project structure for 'lab_2.1' located at 'C:\Users\User\Desktop\lab_2.1'. The file 'user.py' is selected. The main editor window shows the code for 'user.py':

```
1 print("What is your name?")
2 nameUser = str(input())
3 print("How old are you?")
4 oldUser = int(input())
5 print("Where are you live?")
6 liveUser = str(input())
7 print("This is", nameUser)
8 print("It is", oldUser)
9 print("(S)he live in", liveUser)
10
```

The bottom 'Run' console shows the execution output:

```
C:\Users\User\Desktop\lab_2.1\venv\Scripts\python.exe C:/Users/User/Desktop/lab_2.1/user.py
What is your name?
Sap
How old are you?
19
Where are you live?
Stavropol'
This is Sap
It is 19
(S)he live in Stavropol'
Process finished with exit code 0
```

Рисунок 1. user.py

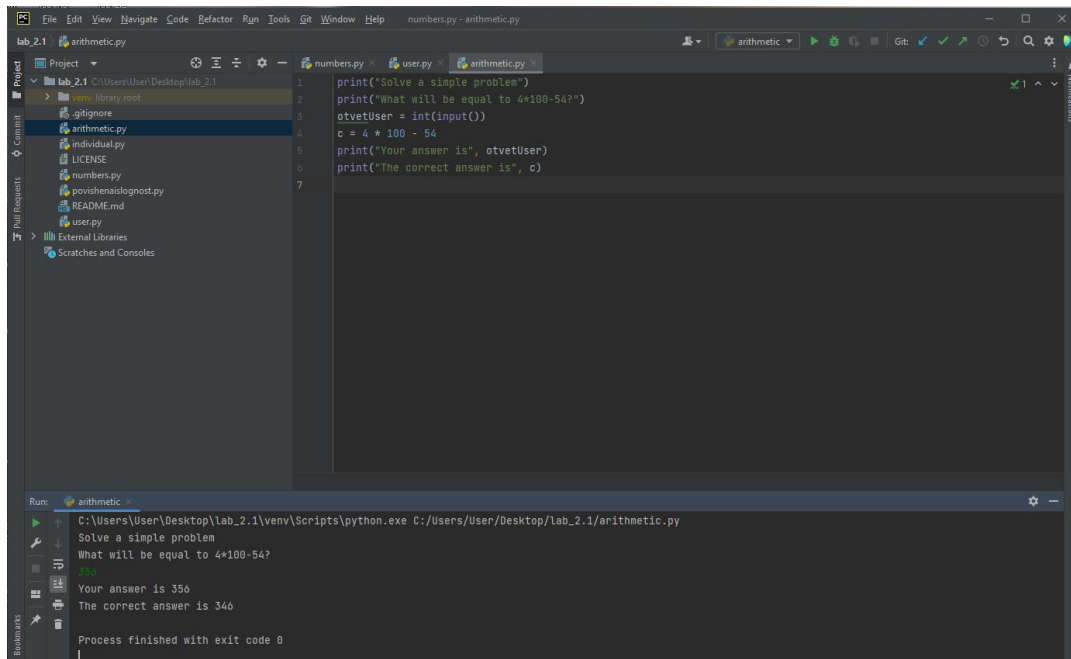


Рисунок 2. arithmetic.py

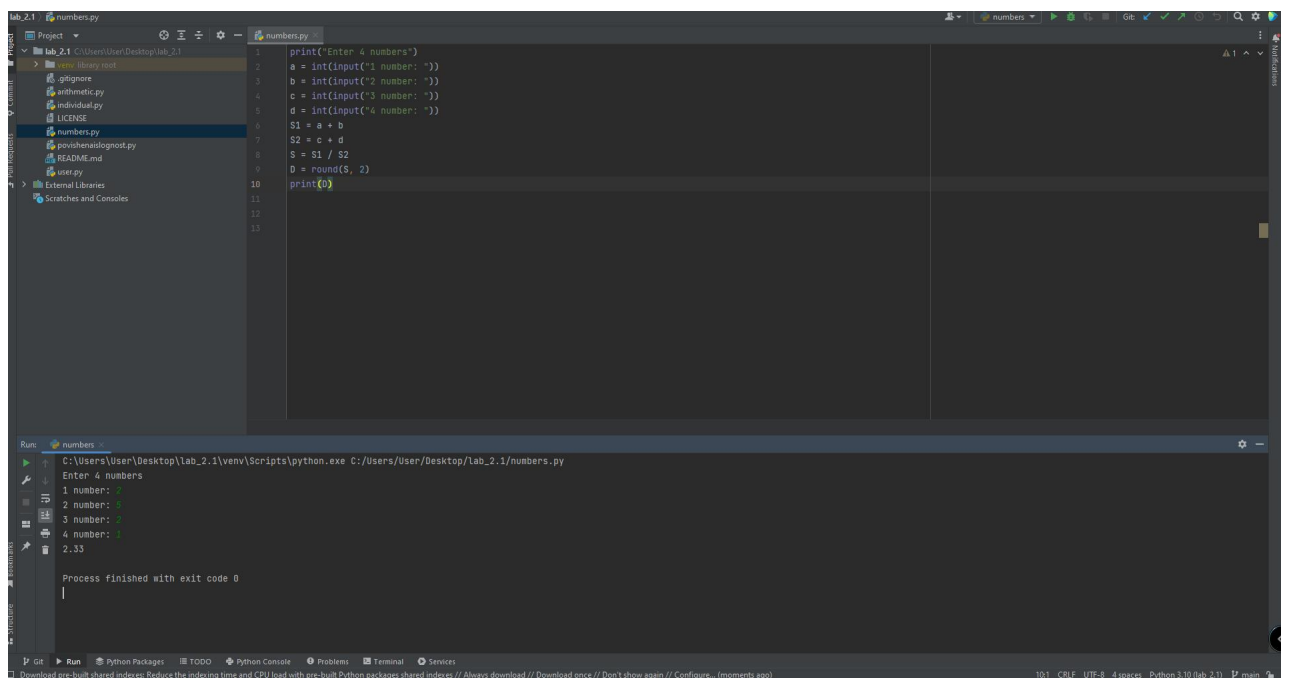


Рисунок 3. numbers.py

7. Напишите программу (файл individual.py) для решения индивидуального задания.

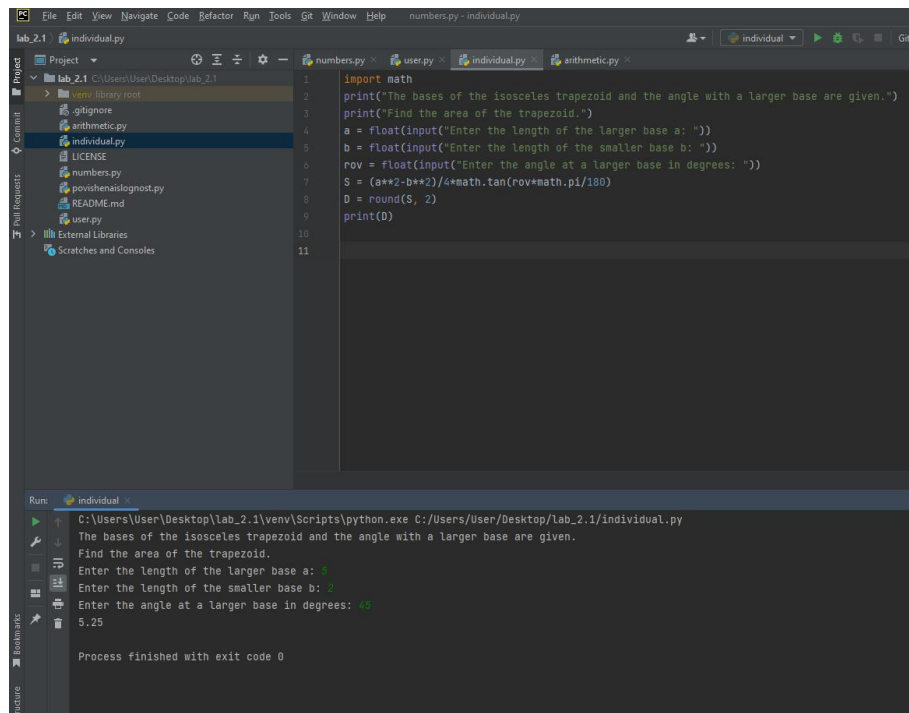


Рисунок 4. individual.py

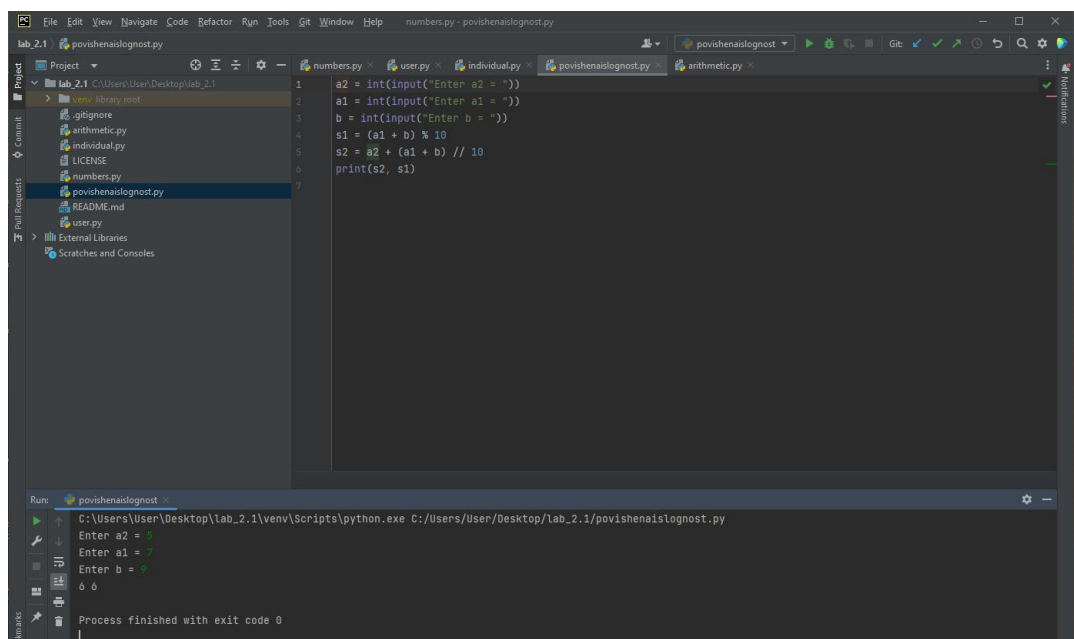


Рисунок 5. Задача повышенной сложности

Ответы на контрольные вопросы:

1. Опишите основные этапы установки Python в Windows и Linux:

Чаще всего интерпретатор Python уже входит в состав дистрибутива.

Windows: Оsn. этапы установки Python на Windows:

1) Скачать дистрибутив с официального сайта;

- 2) Запустить скачанный установочный файл;
- 3) Выбрать способ установки;
- 4) Отметить необходимые опции установки;
- 5) Выбрать место установки;
- 6) Готово.

2. В чем отличие пакета Anaconda от пакета Python, скачиваемого с официального сайта?

Пакет Anaconda содержит версии языка Python 2 и 3, набор наиболее часто используемых библиотек и удобную среду разработки и исполнения, запускаемую в браузере, а также на Anaconda удобнее запускать примеры.

3. Как осуществить проверку работоспособности пакета Anaconda?

Для выполнения проверки работоспособности Anaconda необходимо вначале запустить командный процессор с поддержкой виртуальных окружений Anaconda. В появившейся командной строке необходимо ввести `> jupyter notebook`, в результате чего отобразится процесс загрузки веб-среды Jupyter Notebook, после чего запустится веб-сервер и среда разработки в браузере. Создать ноутбук для разработки, для этого нажать на кнопку New и в появившемся списке выбрать Python. В результате будет создана новая страница в браузере с ноутбуком. Ввести в первой ячейке команду `print("Hello, World!")` и нажать Alt+Enter на компьютере. Ниже ячейки должна появиться соответствующая надпись.

4. Как задать используемый интерпретатор языка Python в IDE PyCharm?

Указать путь до интерпретатора в настройках IDE, для этого:

- 1) Нажмите на шестеренку в верхнем правом углу, выберите "Add..".
- 2) Далее выберите "System Interpreter";
- 3) Нажмите на 3 точки "..." справа от поля с выбором интерпретатора;
- 4) Укажите путь до интерпретатора.

5. Как осуществить запуск программы с помощью IDE PyCharm?

Сочетанием клавиш Shift+F10.

6. В чем суть интерактивного и пакетного режимов работы Python?

Интерактивный. Python можно использовать как калькулятор для различных вычислений, а если дополнительно подключить необходимые математические библиотеки, то по своим возможностям он становится практически равным таким пакетам как Matlab, Octave и т.п.

Пакетный. В этом режиме сначала записывается вся программа, а потом эта программа выполняется полностью.

7. Почему язык программирования Python называется языком динамической типизации?

Т. к. в ЯП Python проверка типа происходит во время выполнения, а не компиляции.

8. Какие существуют основные типы в языке программирования Python?

Типы в ЯП Python:

1. None
2. Логические переменные
3. Числа
4. Списки
5. Строки 6. Бинарные списки
7. Множества
8. Словари

9. Как создаются объекты в памяти? Каково их устройство? В чем заключается процесс объявления новых переменных и работа операции присваивания?

Для того, чтобы объявить и сразу инициализировать переменную необходимо написать её имя, потом поставить знак равенства и значение, с которым эта переменная будет создана. При инициализации переменной, на уровне интерпретатора, создается целочисленный объект, который имеет некоторый идентификатор, значение и тип. Посредством оператора “=” создается ссылка между переменной и объектом.

10. Как получить список ключевых слов в Python?

Список ключевых слов можно получить непосредственно в программе, для этого нужно подключить модуль keyword и воспользоваться командой keyword.kwlist.

11. Каково назначение функций id() и type()?

Функция id() предназначена для получения значения идентичности объекта. С помощью функции type() можно получить тип конкретного объекта.

12. Что такое изменяемые и неизменяемые типы в Python?

К неизменяемым (immutable) типам относятся: целые числа (int), числа с плавающей точкой (float), комплексные числа (complex), логические переменные (bool), кортежи (tuple), строки (str) и неизменяемые множества (frozen set). К изменяемым (mutable) типам относятся: списки (list), множества (set), словари (dict).

13. Чем отличаются операции деления и целочисленного деления?

При целочисленном делении отбрасывается дробная часть от деления чисел, при операции деления дробная часть не отбрасывается.

14. Какие имеются средства в языке Python для работы с комплексными числами?

Для создания комплексного числа можно использовать функцию complex (a, b), в которую, в качестве первого аргумента, передается действительная часть, в качестве второго – мнимая. Либо записать число в виде $a + bj$.

Комплексные числа можно складывать, вычитать, умножать, делить и возводить в степень. У комплексного числа можно извлечь действительную (x.real) и мнимую части (x.imag). Для получения комплексносопряженного числа необходимо использовать метод conjugate().

15. Каково назначение и основные функции библиотеки (модуля) math?

По аналогии с модулем `math` изучите самостоятельно назначение и основные функции модуля `cmath`. Для выполнения математических операций необходим модуль `math`. Осн. операции библиотеки `math`:

`math.ceil(x)` - возвращает ближайшее целое число большее, чем `x`.

`math.fabs(x)` - возвращает абсолютное значение числа.

`math.factorial(x)` - вычисляет факториал `x`.

`math.floor(x)` - возвращает ближайшее целое число меньшее, чем `x`.

`math.exp(x)` - вычисляет $e^{**}x$.

`math.log2(x)` - логарифм по основанию 2.

`math.log10(x)` - логарифм по основанию 10.

`math.log(x[, base])` - по умолчанию вычисляет логарифм по основанию

`e`, дополнительно можно указать основание логарифма. `math.pow(x, y)`

- вычисляет значение `x` в степени `y`.

`math.sqrt(x)` - корень квадратный от `x`.

`math.cos(x)` - косинус от `x`. `math.sin(x)` - синус от `x`.

`math.tan(x)` - тангенс от `x`.

`math.acos(x)` - арккосинус от `x`.

`math.asin(x)` - арксинус от `x`.

`math.atan(x)` - арктангенс от `x`.

`math.pi` - число пи.

`math.e` - число `e`.

16. Каково назначение именных параметров `sep` и `end` в функции `print()`?

Через параметр `sep` можно указать отличный от пробела разделитель строк. Параметр `end` позволяет указывать, что делать, после вывода строки.

17. Каково назначение метода `format()`? Какие еще существуют средства для форматирования строк в Python?

Примечание: в дополнение к рассмотренным средствам изучите самостоятельно работу с f-строками в Python. Форматирование может выполняться в так называемом старом стиле или с помощью строкового

метода `format`. Символы `%s`, `%d`, `%f` подставляются значения переменных. Буквы `s`, `d`, `f` обозначают типы данных – строку, целое число, вещественное число.

18. Каким образом осуществить ввод с консоли значения целочисленной и вещественной переменной в языке Python?

Указать перед `input` тип данных: `int(input())`.

Вывод: исследовал процесс установки и базовые возможности языка программирования Python версии 3.