

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего
образования
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ»**

**Кафедра
инфокоммуникаций
Институт цифрового
развития**

ОТЧЁТ
по лабораторной работе №4
Дисциплина: «Основы программной инженерии»
Тема: «Основы языка Python»

Выполнил: студент 2
курса группы Пиж-б-о-
21-1
Рязанцев Матвей
Денисович

Ставрополь 2022

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

Выполнение работы

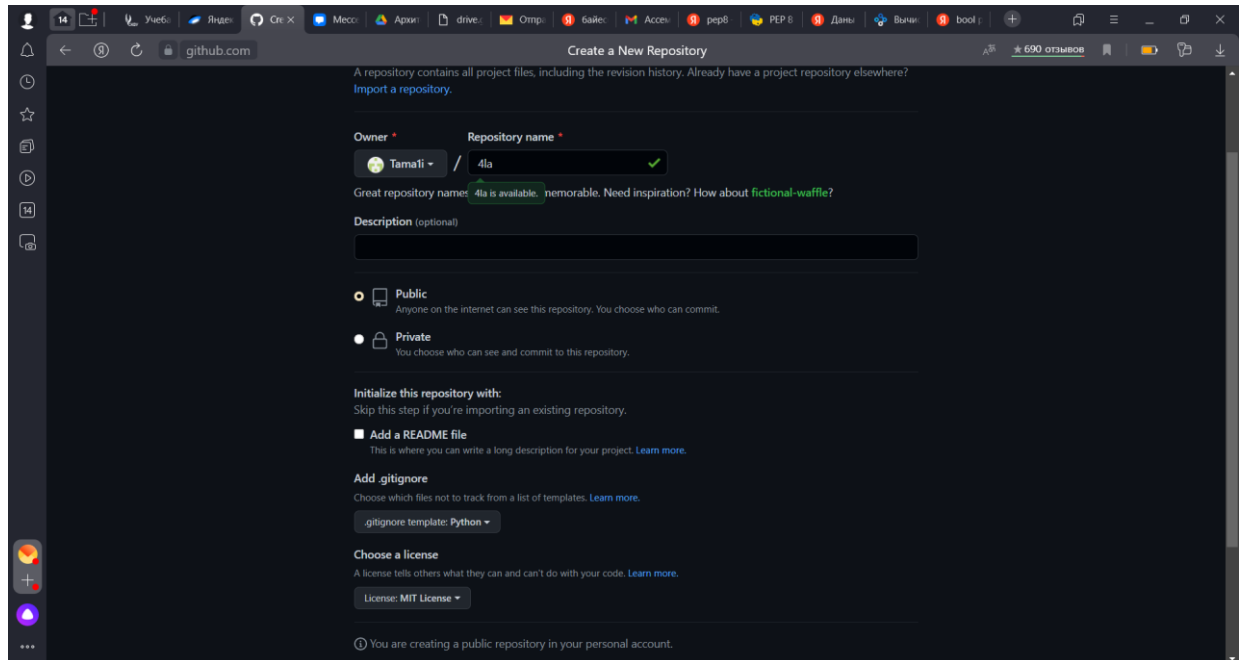


Рисунок 1 – Создание репозитория

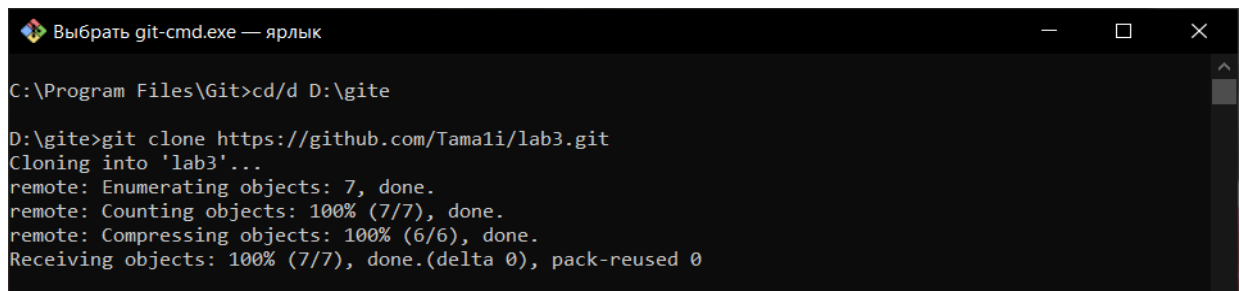


Рисунок 2 – Клонирование репозитория

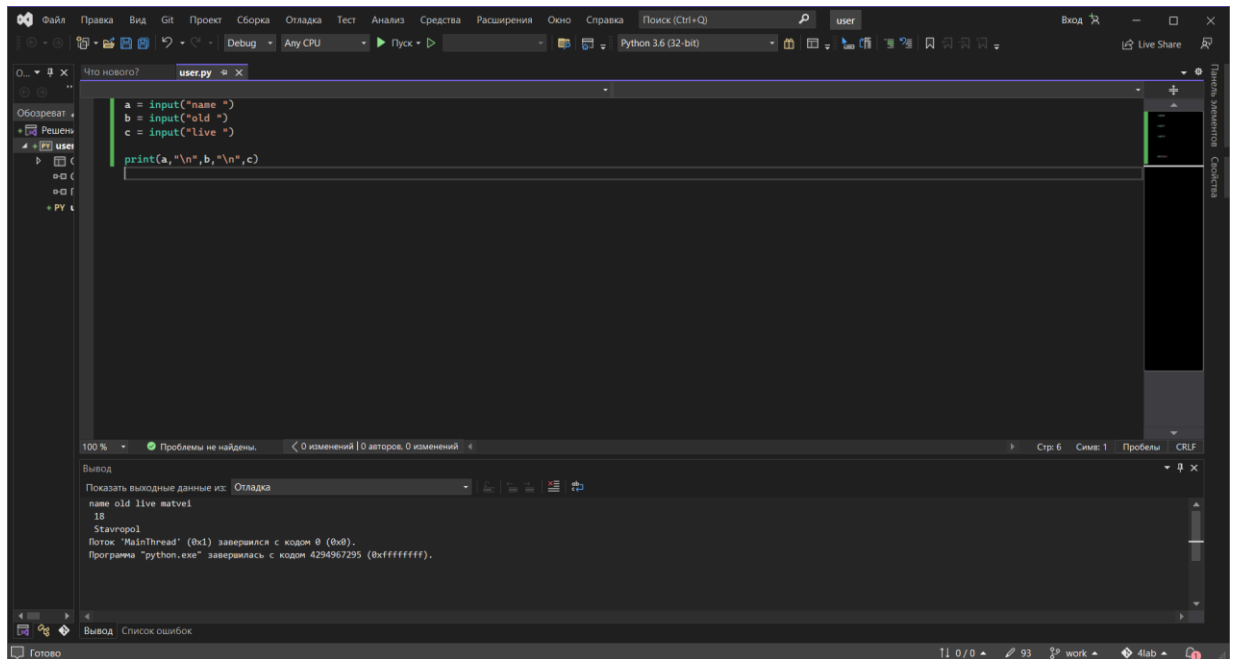


Рисунок 3 – программа user

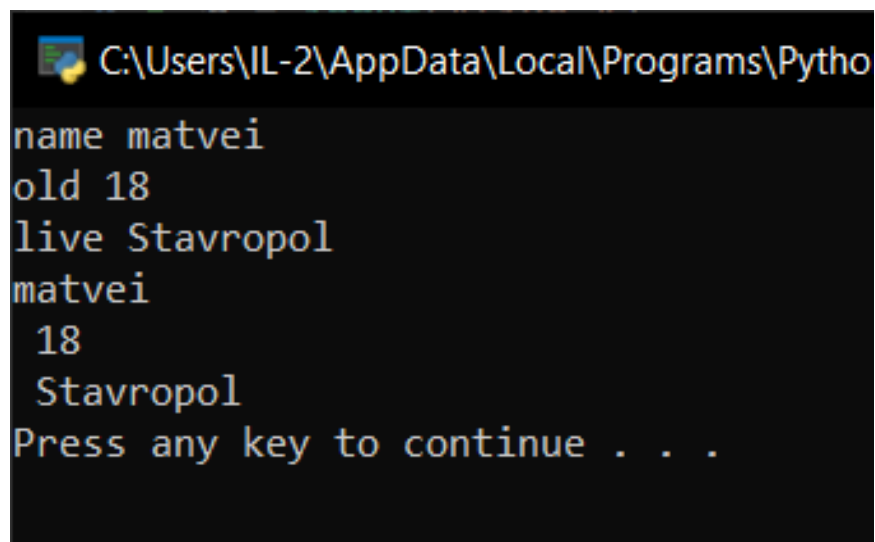


Рисунок 4 – результат работы программы user

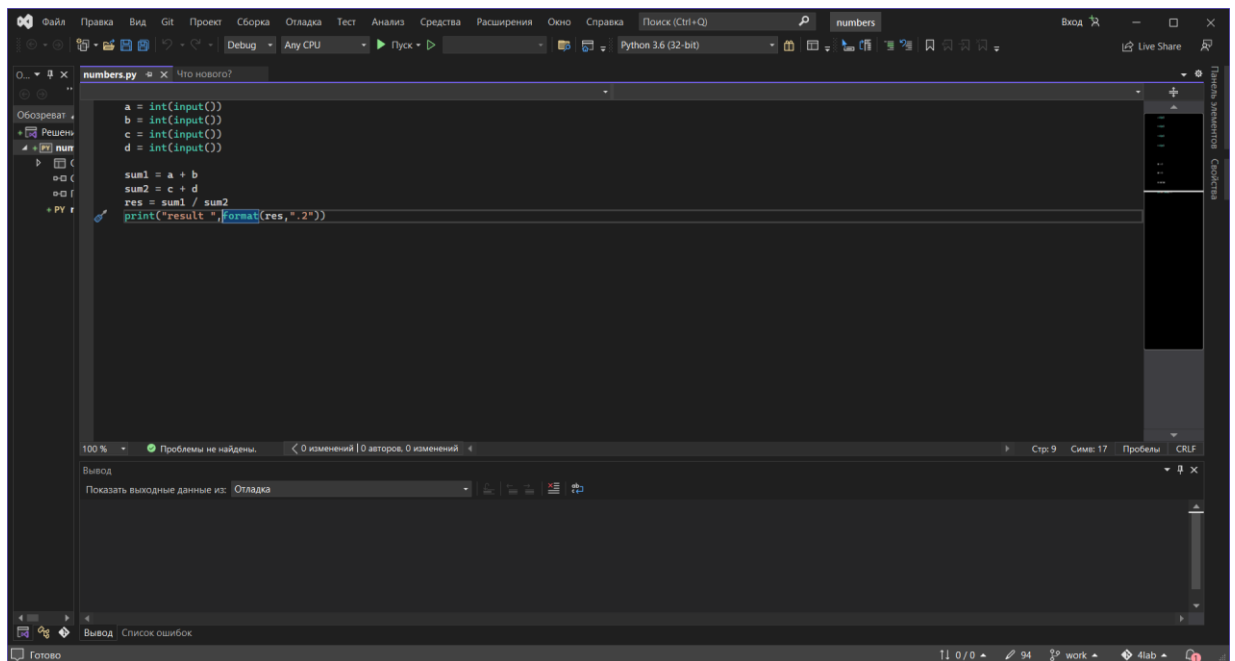


Рисунок 5 – программа numbers

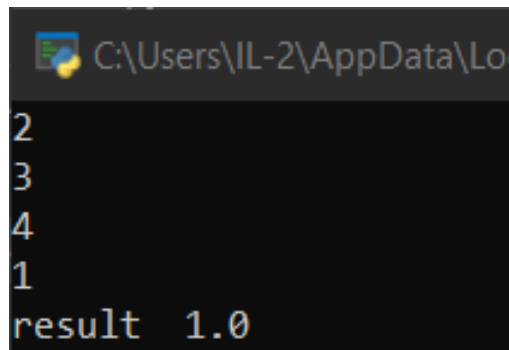


Рисунок 6 – результат работы программы numbers

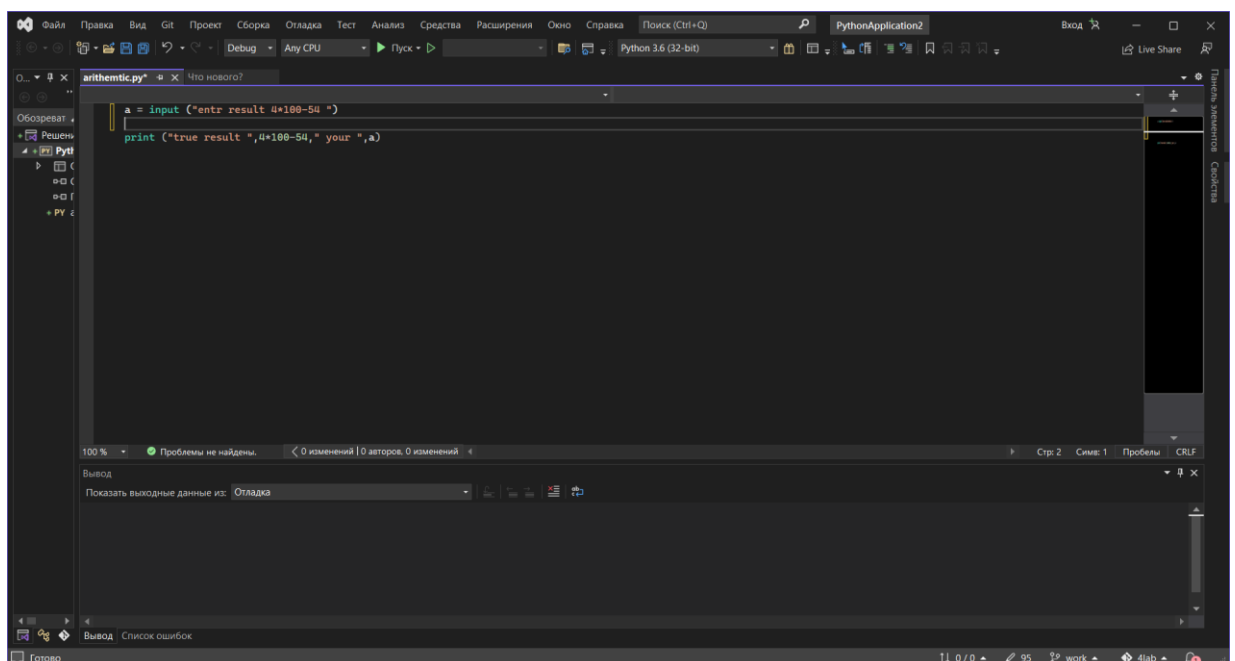
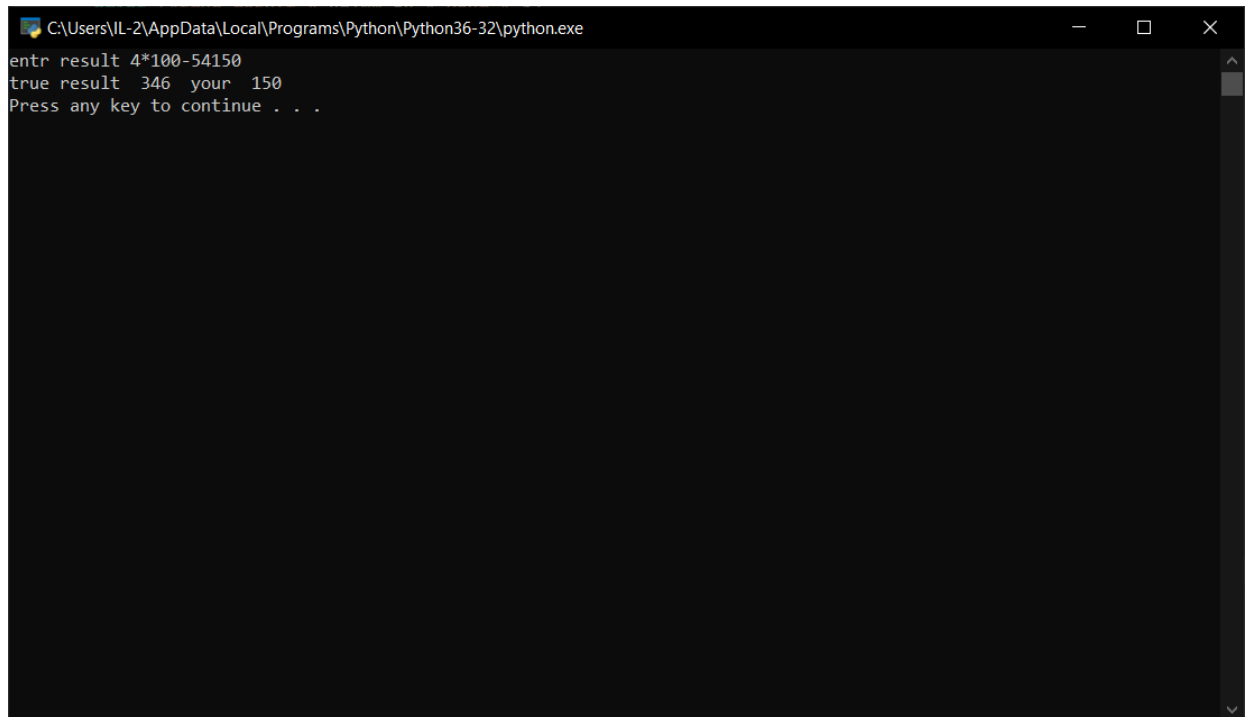
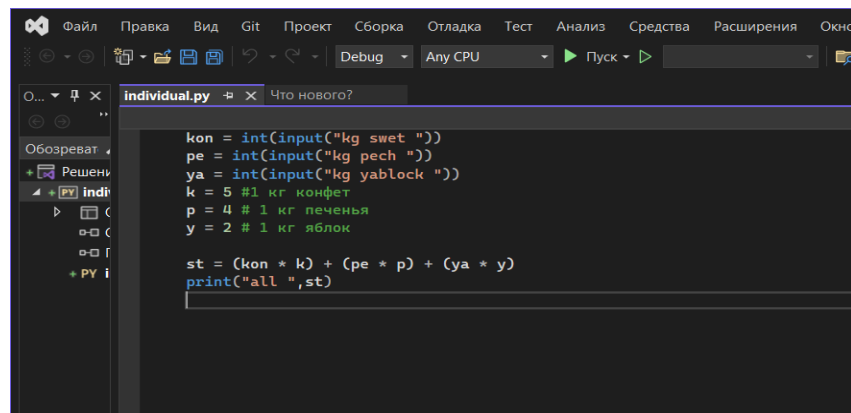


Рисунок 7 – программа arichmetic



```
C:\Users\IL-2\AppData\Local\Programs\Python\Python36-32\python.exe
entr result 4*100-54150
true result 346 your 150
Press any key to continue . . .
```

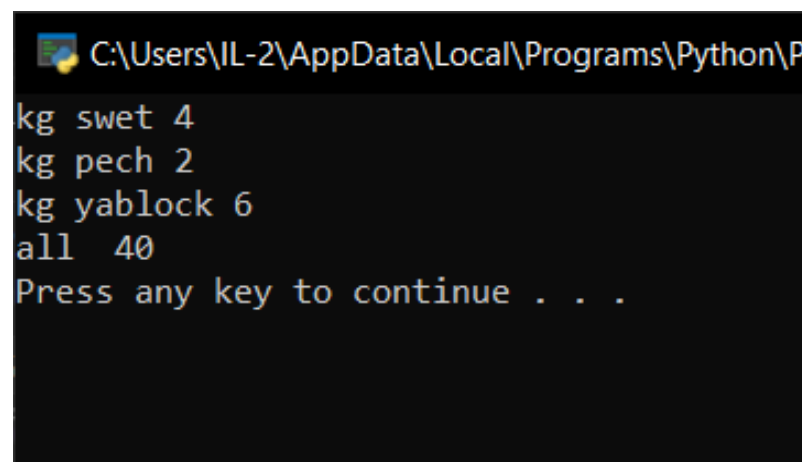
рисунок 8 – результат работы программы arichmetic



```
individual.py
kon = int(input("kg swet "))
pe = int(input("kg pech "))
ya = int(input("kg yablock "))
k = 5 #1 кг конфет
p = 4 # 1 кг печенья
y = 2 # 1 кг яблок

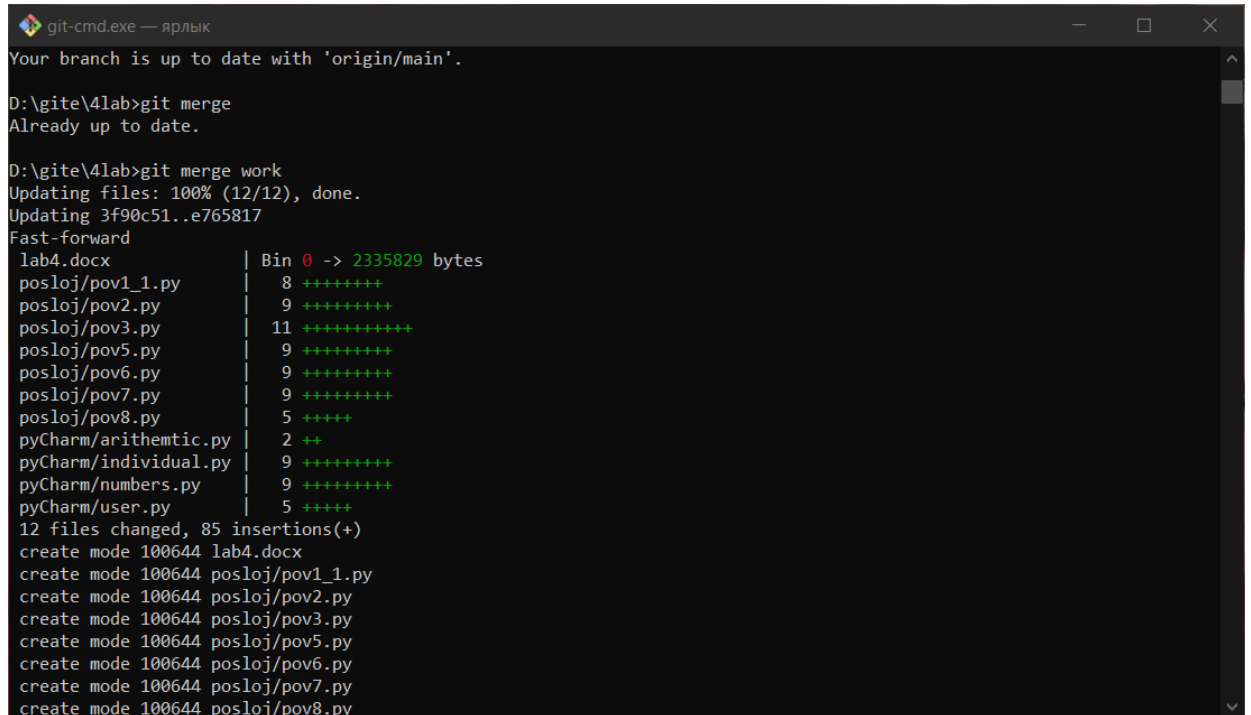
st = (kon * k) + (pe * p) + (ya * y)
print("all ",st)
```

Рисунок 9 – код программы индивидуальное задание



```
C:\Users\IL-2\AppData\Local\Programs\Python\Python36-32\python.exe
kg swet 4
kg pech 2
kg yablock 6
all 40
Press any key to continue . . .
```

Рисунок 10 – результат работы программы

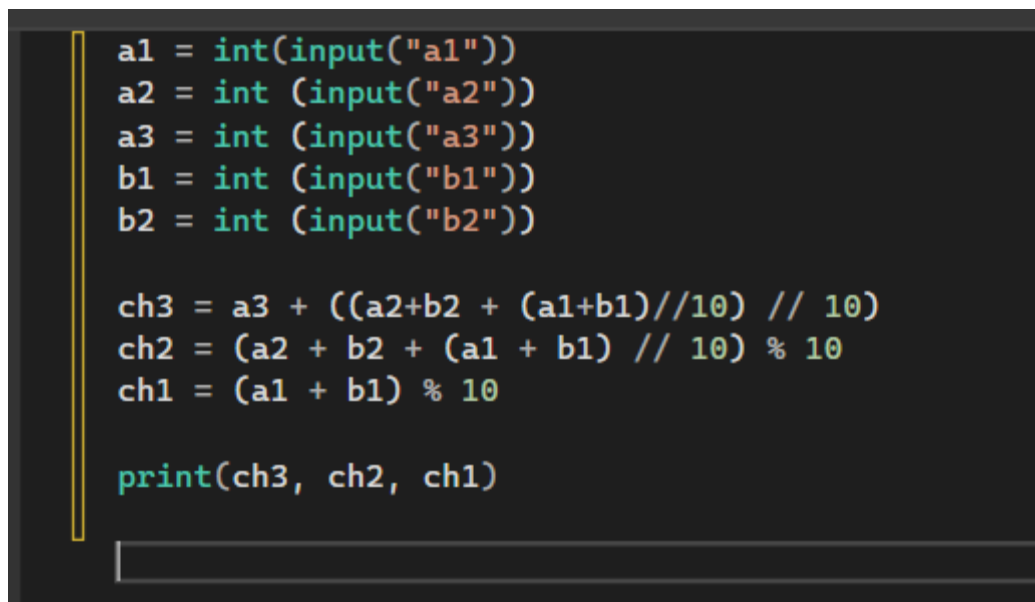


```
git-cmd.exe — ярлык
Your branch is up to date with 'origin/main'.

D:\gite\4lab>git merge
Already up to date.

D:\gite\4lab>git merge work
Updating files: 100% (12/12), done.
Updating 3f90c51..e765817
Fast-forward
 lab4.docx                | Bin 0 -> 2335829 bytes
 posloj/pov1_1.py         | 8 ++++++
 posloj/pov2.py           | 9 ++++++
 posloj/pov3.py           | 11 ++++++++
 posloj/pov5.py           | 9 ++++++
 posloj/pov6.py           | 9 ++++++
 posloj/pov7.py           | 9 ++++++
 posloj/pov8.py           | 5 +++++
 pyCharm/arithmetic.py    | 2 ++
 pyCharm/individual.py    | 9 ++++++
 pyCharm/numbers.py       | 9 ++++++
 pyCharm/user.py          | 5 +++++
12 files changed, 85 insertions(+)
create mode 100644 lab4.docx
create mode 100644 posloj/pov1_1.py
create mode 100644 posloj/pov2.py
create mode 100644 posloj/pov3.py
create mode 100644 posloj/pov5.py
create mode 100644 posloj/pov6.py
create mode 100644 posloj/pov7.py
create mode 100644 posloj/pov8.py
```

Рисунок 11 – слияние веток work и main



```
a1 = int(input("a1"))
a2 = int (input("a2"))
a3 = int (input("a3"))
b1 = int (input("b1"))
b2 = int (input("b2"))

ch3 = a3 + ((a2+b2 + (a1+b1)//10) // 10)
ch2 = (a2 + b2 + (a1 + b1) // 10) % 10
ch1 = (a1 + b1) % 10

print(ch3, ch2, ch1)
```

Рисунок 12- код программы задания повышенной сложности

```
a1 = int(input("a1 "))
a2 = int (input("a2 "))
a3 = int (input("a3 "))
b1 = int (input("b1 "))
b2 = int (input("b2 "))

C:\Users\IL-2\AppData\Local\Programs\Python\Python36-32\python.exe
a1 3
a2 2
a3 1
b1 2
b2 1
1 3 5
Press any key to continue . . .
```

Рисунок 13 – результат работы программы

Вывод: в ходе выполнения работы исследован процесс установки и базовых возможностей языка Python

Контрольные вопросы

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

Для установки интерпретатора Python первое, что нужно сделать – это скачать дистрибутив. Загрузить его можно с официального сайта, перейдя по ссылке <https://www.python.org/downloads/>.

2. Запустить скачанный установочный файл.
3. Выбрать способ установки.
4. Отметить необходимые опций установки
5. Выбирать место установки

При установке для Linux, в случае ошибки необходимо либо собрать

Python из исходников, либо взять из репозитория. Для установки из репозитория в Ubuntu воспользуйтесь командой «`sudo apt-get install python3`»

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

3. Как осуществить проверку работоспособности пакета Anaconda?

В Windows это можно сделать, выбрав следующий пункт главного меню системы Пуск Anaconda3 (64-bit) Anaconda Prompt. 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.

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.

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, выберете "Add..".
- 2) Далее выберете "System Interpreter";
- 3) Нажмите на 3 точки "..." справа от поля в выбор интерпретатора;
- 4) Укажите путь до интерпретатора

5. Как осуществить запуск программы с помощью IDE PyCharm?
Shift+F10

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

В интерактивном.

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.

В проектном.

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

7. Почему язык программирования Python называется языком динамической типизации?

В нем проверка типа происходит во время выполнения, а не компиляции

8. Какие существуют основные типы в языке программирования 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

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?

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 комплексно сопряжённого 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`.

`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 π .

`math.e` - число e .

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

`Sep()` устанавливает отличный от пробела разделитель строк.

`End()` указывает, что делать, после вывода строки (по умолчанию стоит переход на новую строку)

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())`.

`e_ветки>`.