

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

Кафедра инфокоммуникаций

«Основы ветвления Git»

Отчет по лабораторной работе № 2.8

по дисциплине «Основы программной инженерии»

Выполнил студент группы ПИЖ-б-о-21-1

Образцова М. Д.. «14» декабря 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____

(подпись)

Ставрополь 2022

МЕТОДИКА И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Создала репозиторий в GitHub «OPI_LR_11» в который добавила .gitignore для работы с IDE PyCharm с ЯП Python, выбрала лицензию MIT.

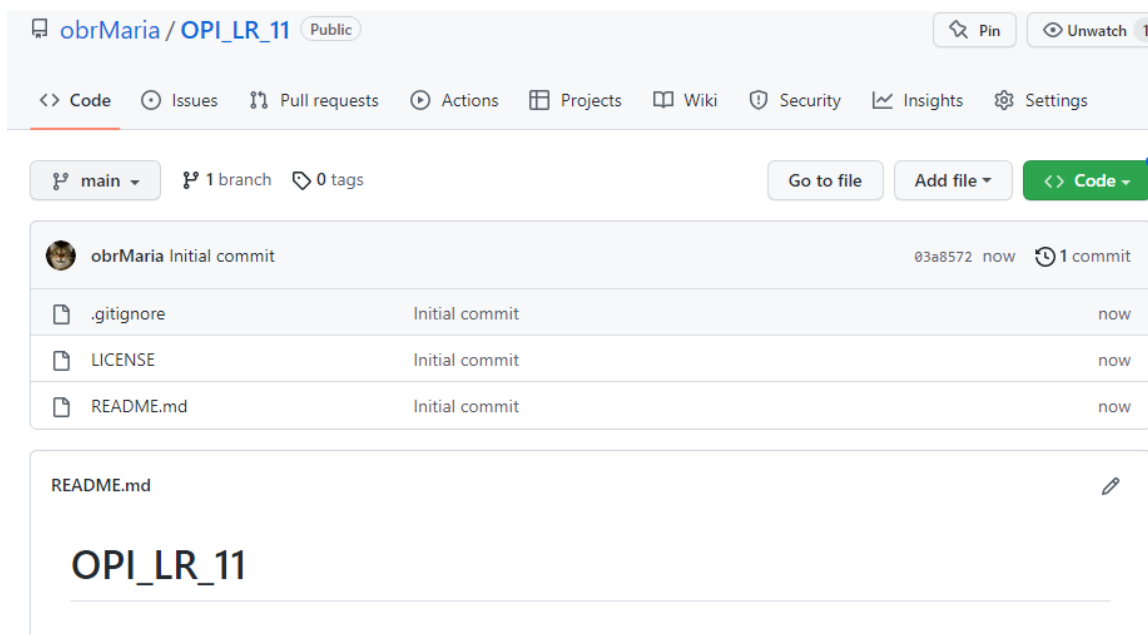


Рисунок 1 – Создание репозитория

```
M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_11 (main)
$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/M/desktop/OPI_LR_11/.git/hooks]

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_11 (develop)
$
```

Рисунок 2 – Организация репозитория в соответствии с моделью ветвления
git-flow

```
M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_11 (develop)
$ git add .

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_11 (develop)
$ git commit -m "gitignore"
[develop a14dbff] gitignore
1 file changed, 140 insertions(+), 2 deletions(-)

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_11 (develop)
$
```

Рисунок 3 – Изменение .gitignore

Проработаны примеры лабораторной работы. Зафиксированы изменения.

```
primer x
add - добавить работника,
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - вывести список команд;
exit - завершить работу с программой.

>>> add
Фамилия и инициалы: образцова мд
Должность: студент
Год поступления: 2021
>>> add
Фамилия и инициалы: коновалова вн
Должность: дизайнер
Год поступления: 2017
>>> add
Фамилия и инициалы: кучеренко сю
Должность: фотограф
Год поступления: 2006
>>> select
>>> Неизвестная команда select

Неизвестная команда
>>> select 10

+-----+-----+-----+-----+
| № |          Ф.И.О.          | Должность | Год |
+-----+-----+-----+-----+
|  1 | кучеренко сю             | фотограф  | 2006 |
+-----+-----+-----+-----+
>>> list

+-----+-----+-----+-----+
| № |          Ф.И.О.          | Должность | Год |
+-----+-----+-----+-----+
|  1 | коновалова вн           | дизайнер  | 2017 |
+-----+-----+-----+-----+
|  2 | кучеренко сю            | фотограф  | 2006 |
+-----+-----+-----+-----+
|  3 | образцова мд            | студент   | 2021 |
+-----+-----+-----+-----+
>>> exit
```

Run | TODO | Problems | Debug | Terminal | Python Packages | Python Console

Рисунок 4 – Результат выполнения программы

Общие задания:

1) Решить следующую задачу: основная ветка программы, не считая заголовков функций, состоит из двух строки кода. Это вызов функции `test()` и инструкции `if __name__ == '__main__':`. В ней запрашивается на ввод целое число. Если оно положительное, то вызывается функция `positive()`, тело которой содержит команду вывода на экран слова "Положительное". Если число отрицательное, то вызывается функция `negative()`, ее тело содержит выражение вывода на экран слова "Отрицательное".

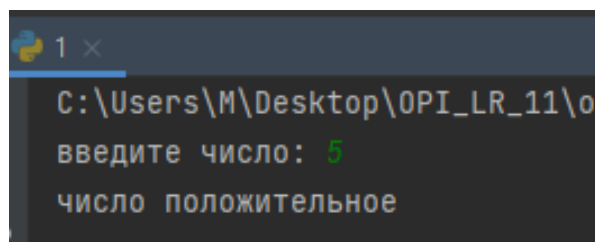
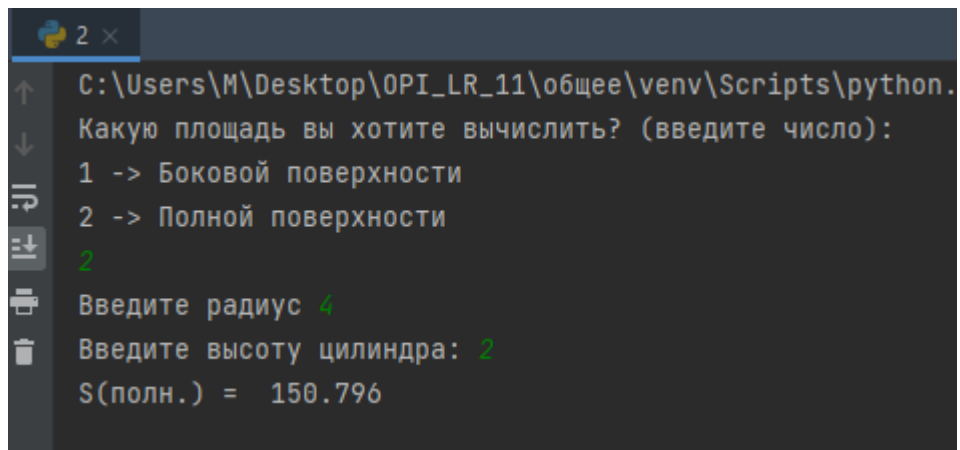


Рисунок 4 – Результат выполнения программы 1

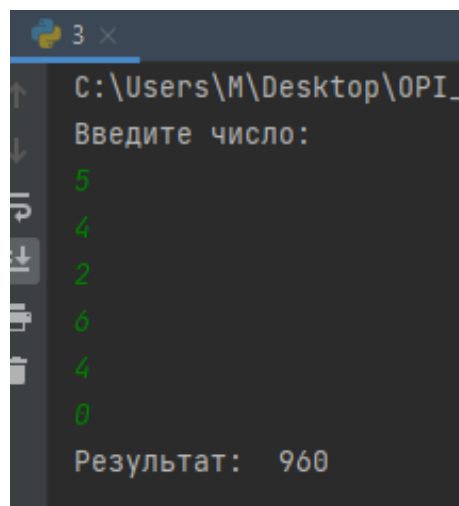
2) Решите следующую задачу: в основной ветке программы вызывается функция `cylinder()`, которая вычисляет площадь цилиндра. В теле `cylinder()` определена функция `circle()`, вычисляющая площадь круга по формуле πr^2 . В теле `cylinder()` у пользователя спрашивается, хочет ли он получить только площадь боковой поверхности цилиндра, которая вычисляется по формуле $2\pi r h$, или полную площадь цилиндра. В последнем случае к площади боковой поверхности цилиндра должен добавляться удвоенный результат вычислений функции `circle()`.



```
C:\Users\M\Desktop\OPI_LR_11\общее\venv\Scripts\python.  
Какую площадь вы хотите вычислить? (введите число):  
1 -> Боковой поверхности  
2 -> Полной поверхности  
2  
Введите радиус 4  
Введите высоту цилиндра: 2  
S(полн.) = 150.796
```

Рисунок 5 – Результат выполнения программы 2

3) Решите следующую задачу: напишите функцию, которая считывает с клавиатуры числа и перемножает их до тех пор, пока не будет введен 0. Функция должна возвращать полученное произведение. Вызовите функцию и выведите на экран результат ее работы.



```
C:\Users\M\Desktop\OPI_LR_11\общее\venv\Scripts\python.  
Введите число:  
5  
4  
2  
6  
4  
0  
Результат: 960
```

Рисунок 6 – Результат выполнения программы 3

4) Решите следующую задачу: напишите программу, в которой определены следующие четыре функции:

1. Функция `get_input()` не имеет параметров, запрашивает ввод с клавиатуры и возвращает в основную программу полученную строку.

2. Функция `test_input()` имеет один параметр. В теле она проверяет, можно ли переданное ей значение преобразовать к целому числу. Если можно, возвращает логическое `True`. Если нельзя – `False`.

3. Функция `str_to_int()` имеет один параметр. В теле преобразовывает переданное значение к целочисленному типу. Возвращает полученное число.

4. Функция `print_int()` имеет один параметр. Она выводит переданное значение на экран и ничего не возвращает.

В основной ветке программы вызовите первую функцию. То, что она вернула, передайте во вторую функцию. Если вторая функция вернула `True`, то те же данные (из первой функции) передайте в третью функцию, а возвращенное третьей функцией значение – в четвертую.

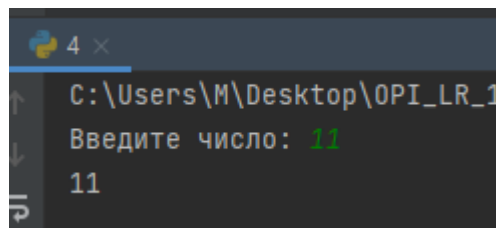
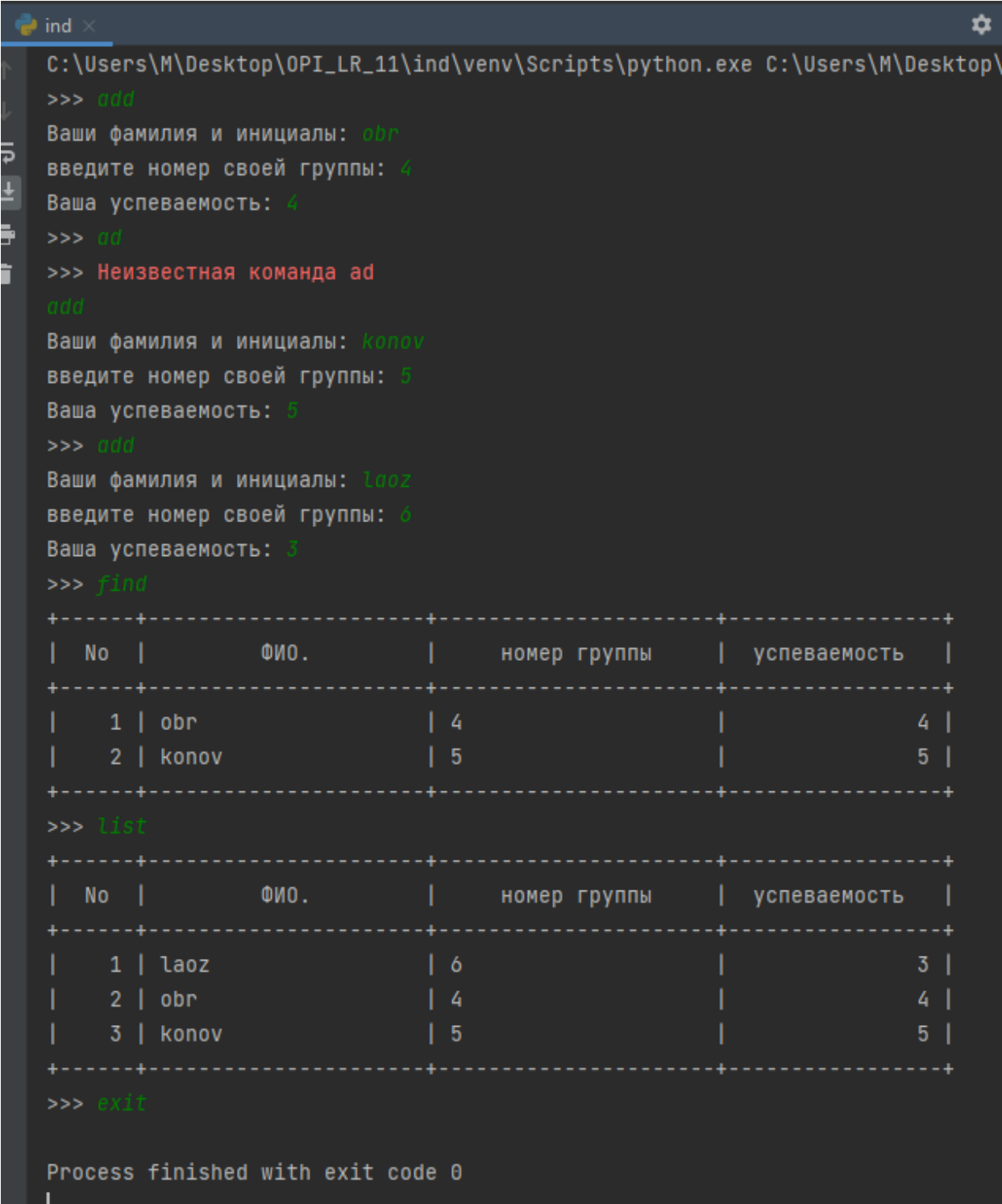


Рисунок 7 – Результат выполнения программы 4

Индивидуальное задание

Решить индивидуальное задание лабораторной работы 2.6, оформив каждую команду в виде отдельной функции.



```
ind x
C:\Users\M\Desktop\OPI_LR_11\ind\venv\Scripts\python.exe C:\Users\M\Desktop\
>>> add
Ваши фамилия и инициалы: obr
введите номер своей группы: 4
Ваша успеваемость: 4
>>> ad
>>> Известная команда ad
add
Ваши фамилия и инициалы: konov
введите номер своей группы: 5
Ваша успеваемость: 5
>>> add
Ваши фамилия и инициалы: laoz
введите номер своей группы: 6
Ваша успеваемость: 3
>>> find
+-----+-----+-----+-----+
| No |      ФИО.      | номер группы | успеваемость |
+-----+-----+-----+-----+
|  1 | obr             | 4            | 4            |
|  2 | konov           | 5            | 5            |
+-----+-----+-----+-----+
>>> list
+-----+-----+-----+-----+
| No |      ФИО.      | номер группы | успеваемость |
+-----+-----+-----+-----+
|  1 | laoz           | 6            | 3            |
|  2 | obr            | 4            | 4            |
|  3 | konov          | 5            | 5            |
+-----+-----+-----+-----+
>>> exit

Process finished with exit code 0
```

Рисунок 8 – Результат выполнения индивидуального задания

Контрольные вопросы:

1. Каково назначение функций в языке программирования Python?

Главной задачей функций в Python, как и в других языках программирования, является сокращение объёма кода и его структуризация. В функции, как правило, выносятся те части кода, которые выполняются в программе многократно.

2. Каково назначение операторов `def` и `return`?

Оператор `def` необходим для определения функции. После него идёт название самой функции, передаваемые в функцию параметры и само тело функции. Оператор `return` служит для возвращения результата выполнения функции в основную программу, где эта функция была вызвана.

3. Каково назначение локальных и глобальных переменных при написании функций Python?

Локальные переменные существуют только внутри функции. В другой части программы как-либо вызывать или изменить их невозможно.

Глобальные напротив – существуют во всей программе.

4. Как вернуть несколько значений из функции Python?

После оператора `return` необходимо записать все возвращаемые переменные через запятую, а при вызове функции нужно задать необходимое количество переменных. Куда будут возвращены параметры.

5. Какие существуют способы передачи значений в функцию?

По ссылке и по значению.

6. Как задать значение аргументов функции по умолчанию?

Нужно в скобках передаваемых параметров присвоить им значение.

7. Каково назначение lambda-выражений в языке Python?

Lambda-выражения – это небольшие функции, которые вызываются в программе один раз.

8. Как осуществляется документирование кода согласно PEP257?

Если пояснение функции содержит одну строку, то достаточно двух кавычек с каждой стороны строки. Пример: `"""Пояснение"""`. Если это многострочное пояснение, то необходимо три кавычки с каждой стороны. Пояснение находится в теле функции, сразу после её объявления.

9. В чем особенность однострочных и многострочных форм строк документации?