

**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 № 2.4

по дисциплине «Основы программной инженерии»

Выполнил студент группы ПИЖ-б-о-21-1

Образцова М. Д. .«22» ноября 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_7» в который добавила .gitignore для работы с IDE PyCharm с ЯП Python, выбрала лицензию MIT.

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .gitignore

no changes added to commit (use "git add" and/or "git commit -a")

1@DESKTOP-A8I4D4M MINGW64 ~/desktop/OPI_LR_7 (main)
$ git add .

1@DESKTOP-A8I4D4M MINGW64 ~/desktop/OPI_LR_7 (main)
$ git commit -m "gitignore"
[main 5802864] gitignore
1 file changed, 140 insertions(+), 2 deletions(-)
```

Рисунок 1– Изменение .gitignore

```
1@DESKTOP-A8I4D4M MINGW64 ~/desktop/OPI_LR_7 (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/1/desktop/OPI_LR_7/.git/hooks]

1@DESKTOP-A8I4D4M MINGW64 ~/desktop/OPI_LR_7 (develop)
```

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

2. Проработаны примеры лабораторной работы. Для каждого примера создан отдельный модуль языка Python. Зафиксированы изменения в репозитории.

```
if __name__ == '__main__':  
    if len(A) != 0:  
        print(A)  
        print(0)
```

Рисунок 2 – Результат выполнения программы

```
C:\Users\M\Desktop  
-1 5 9 8 11  
3
```

Рисунок 3 – Результат выполнения программы

Индивидуальные задания

(21.) В списках A, G, F содержатся оценки учащихся по алгебре, геометрии и физике соответственно. Определить, по какому предмету лучше успеваемость.

Решить индивидуальное задание как с использованием циклов, так и с использованием List Comprehensions.

```
1 x  
C:\Users\M\Desktop\OPI_LR_7\PyCharm\ind\venv\Scripts  
оценки по алгебре 4 4 5  
оценки по геометрии 5 3  
оценки по физике 3 4 5 5 5  
по физике лучшая успеваемость = 4.4
```

Рисунок 4 – Вывод программы индивидуального задания 1

(2.) В списке, состоящем из вещественных элементов, вычислить:

- Сумму положительных элементов списка;

- Произведение элементов списка, расположенных между максимальным по модулю и минимальным по модулю элементами.
- Упорядочить элементы списка по убыванию.

```

C:\Users\M\Desktop\OPI_LR_7\PyCharm\ind\venv\
введите элементы 9 2 5 -3 1 12 8 0
сумма положительных элементов равна 37
произведение между мин и макс эл-ми равно 8
[12, 9, 8, 5, 2, 1, 0, -3]

Process finished with exit code 0

```

Рисунок 5 – Вывод программы индивидуального задания 1

Для сортировки допускается использовать метод `sort` с заданным параметром `key` (<https://docs.python.org/3/howto/sorting.html>) и объединение нескольких списков.

ВОПРОСЫ ДЛЯ ЗАЩИТЫ РАБОТЫ

1. Что такое списки в языке Python?

Список (`list`) – это структура данных для хранения объектов различных типов.

2. Как осуществляется создание списка в Python?

Для создания списка нужно заключить элементы в квадратные скобки.

3. Как организовано хранение списков в оперативной памяти?

Список является изменяемым типом данных. При его создании в памяти резервируется область, которую можно условно назвать некоторым “контейнером”, в котором хранятся ссылки на другие элементы данных в памяти. В отличие от таких типов данных как число или строка, содержимое

“контейнера” списка можно менять.

4. Каким образом можно перебрать все элементы списка?

`for elem in my_list:`

5. Какие существуют арифметические операции со списками?

`+, *`

6. Как проверить есть ли элемент в списке?

Для того, чтобы проверить, есть ли заданный элемент в списке Python необходимо использовать оператор `in`.

7. Как определить число вхождений заданного элемента в списке?

`list.count('элемент')`

8. Как осуществляется добавление (вставка) элемента в список?

Метод `insert` можно использовать, чтобы вставить элемент в список.

9. Как выполнить сортировку списка?

`list.sort()`

10. Как удалить один или несколько элементов из списка?

Удалить элемент можно, написав его индекс в методе `pop`.

11. Что такое списковое включение и как с его помощью осуществлять обработку списков?

List Comprehensions чаще всего на русский язык переводят как абстракция списков или списковое включение, является частью синтаксиса языка, которая предоставляет простой способ построения списков.

12. Как осуществляется доступ к элементам списков с помощью срезов?

`list[<начало среза>:<конец среза>:<шаг>]`

13. Какие существуют функции агрегации для работы со списками?

Для работы со списками Python предоставляет следующие функции:

- `len(L)` - получить число элементов в списке `L` .
- `min(L)` - получить минимальный элемент списка `L` .
- `max(L)` - получить максимальный элемент списка `L` .
- `sum(L)` - получить сумму элементов списка `L` , если список `L` содержит только числовые значения

14. Как создать копию списка?

Для создания копии списка необходимо использовать либо метод `copy`, либо использовать оператор среза

15. Самостоятельно изучите функцию `sorted` языка Python. В чем ее отличие от метода `sort` списков?

Отличие заключается в том, что метод `list.sort()` определён только для списков, в то время как `sorted()` работает со всеми итерируемыми объектами.