Московский государственный технический университет им. Н.Э. Баумана.

Факультет «Информатика и управление»

Кафедра ИУ5. Курс «Парадигмы и конструкции языков программирования»

Отчет по рубежному контролю №1

Выполнил: Сорокин Михаил
Группа ИУ5-33Б
Вариант 15 В

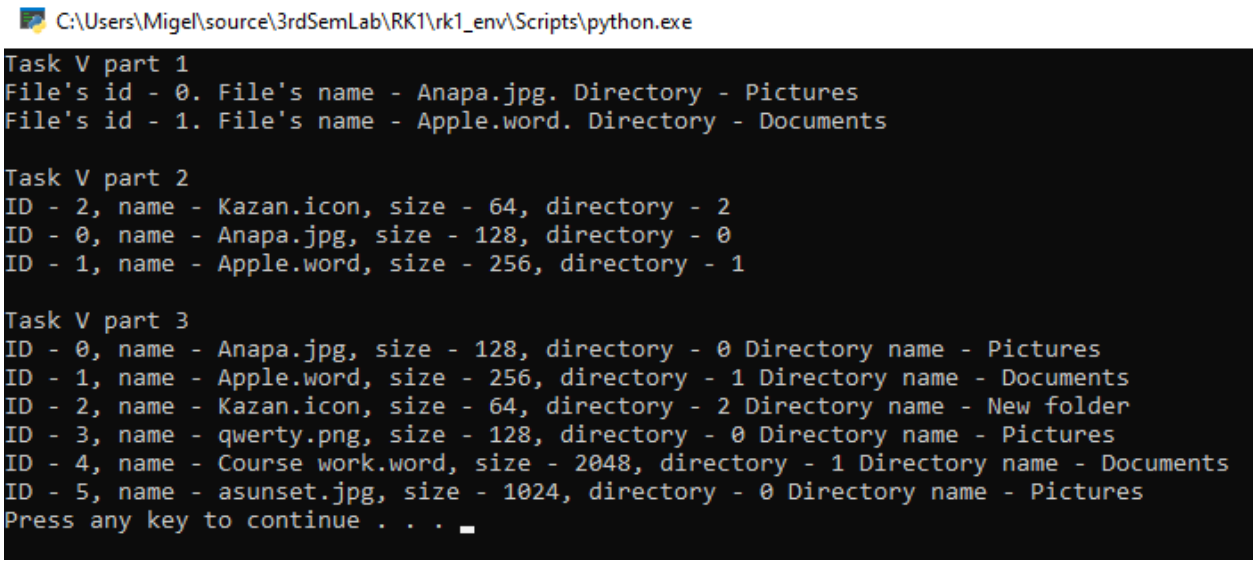
Проверил: Гапанюк Юрий Евгеньевич
преподаватель каф. ИУ5

Подпись и дата: 23.10.2024

Подпись и дата:

Москва, 2024 г.

Принт-скрин выполнения программы



Листинг

Пакет Var15 модуль files.py

```
class File:
    def __init__(self, numb, name, size, dir_id):
        self.__file_id = numb
        self.__name = name
        self.__size = size
        self.__dir_id = dir_id

    @property
    def file_id(self):
        return self.__file_id

    @property
    def name(self):
        return self.__name

    @property
    def dir_id(self):
        return self.__dir_id

    @property
    def size(self):
        return self.__size

    def __repr__(self):
        return ("ID - {}, name - {}, size - {}, directory - {}".format(
            self.__file_id,
            self.__name,
            self.__size,
            self.__dir_id)
        )

class Directory:
    def __init__(self, numb, name):
        self.__dir_id = numb
        self.__name = name

    @property
    def dir_id(self):
```

```
        return self.__dir_id

    @property
    def name(self):
        return self.__name

    def __repr__(self):
        return ("Directory name - {}".format(
            self.__name
        ))

class DirectoryFiles:
    def __init__(self, file_id, dir_id):
        self.__file_id = file_id
        self.__dir_id = dir_id

    @property
    def dir_id(self):
        return self.__dir_id

    @property
    def file_id(self):
        return self.__file_id
```

Исполняемый файл RK1.py

```
from Var15 import files as file_sys

def main():
    # Data
    dir_tuple = (
        file_sys.Directory(0, "Pictures"),
        file_sys.Directory(1, "Documents"),
        file_sys.Directory(2, "New folder"),
    )

    files_tuple = (
        file_sys.File(0, "Anapa.jpg", 128, 0),
        file_sys.File(1, "Apple.word", 256, 1),
        file_sys.File(2, "Kazan.icon", 64, 2),
        file_sys.File(3, "qwerty.png", 128, 0),
        file_sys.File(4, "Course work.word", 2048, 1),
        file_sys.File(5, "asunset.jpg", 1024, 0),
    )

    dirFiles_tuple = (
        file_sys.DirectoryFiles(files_tuple[0].file_id, dir_tuple[0].dir_id),
        file_sys.DirectoryFiles(files_tuple[1].file_id, dir_tuple[1].dir_id),
        file_sys.DirectoryFiles(files_tuple[2].file_id, dir_tuple[2].dir_id),
        file_sys.DirectoryFiles(files_tuple[3].file_id, dir_tuple[0].dir_id),
        file_sys.DirectoryFiles(files_tuple[4].file_id, dir_tuple[1].dir_id),
        file_sys.DirectoryFiles(files_tuple[5].file_id, dir_tuple[0].dir_id),
        )

    # Functionality
    # V - 1
    print("Task V part 1")
    for elem in files_tuple:
        if elem.name[0] == "A":
            print("File's id - {}. File's name - {}. Directory - {}".format(
                elem.file_id,
                elem.name,
                dir_tuple[elem.dir_id].name)
            )

    # V - 2
```

```
    print("\nTask V part 2")
    used = [0] * len(dir_tuple)
    new_list = sorted(files_tuple, key = lambda files_tuple: files_tuple.size)
    #print(new_list)

    def info(file_obj):
        if used[file_obj.dir_id] == 0:
            used[file_obj.dir_id] = 1
            return True
        else:
            return False

    result = filter(info, new_list)
    print(*result, sep = '\n')

    # V - 3
    print("\nTask V part 3")
    dirFiles_list = sorted(dirFiles_tuple, key = lambda dirFiles_tuple: dirFiles_tuple.file_id)
    for elem in dirFiles_list:
        print(files_tuple[elem.file_id], dir_tuple[elem.dir_id])
    # for myself
    #[print(files_tuple[elem.file_id], dir_tuple[elem.dir_id]) for elem in dirFiles_list]
if __name__ == "__main__":
    main()

# Var 15 letter V
```