

Государственное образовательное учреждение высшего профессионального образования



**«Московский государственный технический университет
имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ

Информатика и системы управления

КАФЕДРА

Системы обработки информации и управления

О т ч ё т п о л а б о р а т о р н о й р а б о т е

п о к у р с у

«Разработка интернет-приложений»

Введение в Python

Исполнитель: студентка группы РТ5-51
 Галичий Д.А.

Преподаватель: Гапанюк Ю.Е.

Москва, 2017

Цель работы: ознакомиться с базовым синтаксисом Python.
Дополнительная задача – знакомство с Git и GitHub.

Содержание файла «arr_algs.py»:

```
def minimum(array):
    if len(array) == 0:
        return -1

    min_value = array[0]
    for i in range(len(array)):
        if array[i] < min_value:
            min_value = array[i]
    return min_value


def average(array):
    summ = 0
    for x in array:
        summ += x
    return summ / len(array)


def main():
    arr = [34, 553, 11, 84, 7, 506, 202]
    print(minimum(arr))
    print(average(arr))

main()
```

Функция «minimum (array)» реализует нахождение минимума в массиве. Нахождение среднего арифметического в массиве осуществляется функция «average (array)».

Результат работы программы:

```
7
199.57142857142858
```

Содержание файла «str_algs.py»:

```
def reverse_string(input_string):
    index = len(input_string) - 1
    reversed_string = ""

    for _ in input_string:
```

```

        reversed_string += input_string[index]
        index -= 1
    # return(input_string[::-1])
    return reversed_string

def main():
    str = "hello, world"
    print(reverse_string(str))

main()

```

Функция «reverse_string(input_string)» выполняет переворот строки.
Результат работы программы:
dlrow ,olleh

Содержание файла «dict_algs.py»:

```

def search_employees(emps, child_age):
    employees = []
    for employee in emps:
        for child in employee["children"]:
            if child["age"] > 18:
                employees.append(employee["name"])
    return employees

def main():
    ivan = {
        "name": "ivan",
        "age": 34,
        "children": [ {
            "name": "vasja",
            "age": 12,
        }, {
            "name": "petja",
            "age": 10,
        } ],
    }
    darja = {
        "name": "darja",
        "age": 41,
        "children": [ {
            "name": "kirill",
        } ],
    }

```

```
        "age": 21,  
    }, {  
        "name": "pavel",  
        "age": 15,  
    }],  
}  
emps1 = [ivan, darja]  
child_age1 = 18  
  
print(search_employees(emps1, child_age1))  
  
main()
```

Программа выводит имена сотрудников, у которых есть дети старше 18 лет.

Результат работы программы:
['darja']