

**Министерство науки и высшего образования Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Факультет безопасности информационных технологий**

**Дисциплина:**

«Технологии и методы программирования»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

**Выполнила:**

Чу Ван Доан – Студент группы N3347



\_\_\_\_\_  
(подпись)

**Проверил:**

Ищенко Алексей Петрович

\_\_\_\_\_  
(отметка о выполнении)

\_\_\_\_\_  
(подпись)

Санкт-Петербург

2024 г.

## СОДЕРЖАНИЕ

1 Задание	3
2. Ход работы	3
2. 1 Описание программы	3
2. 2 Выполнение программы	4
2. 2 Код	6
Заключение	11

## **1 Задание**

Разработать простейшую программу, запрашивающую ФИО пользователя и заносщую эту информацию в текстовый файл. Если такое ФИО имеется в файле, то выдавать об этом сообщение. После ввода информации программа должна завершать работу и сообщать пользователю о лимитах ее использования (временнОго или количества запусков). По достижении лимита запусков программа должна предложить пользователю приобрести ее полную версию или деинсталлировать себя. При повторной установке программы, она должна сообщать о своем предыдущем нахождении на этом компьютере и сверяться с прошлыми лимитами пользования (т.е. не давать их суммарно превысить).

Выполняются две версии программы (можно сочетать в одной):

- а) Time-limited (ограничение по времени сделать не более 3 минут, чтобы можно было проследить в момент сдачи достижение лимита).
- б) Start-limited (ограничение на количество запусков тоже должно быть наглядным, например – 4-5).

## **2. Ход работы**

### **2. 1 Описание программы**

Программа состоит из трех основных этапов: установка (install.sh), запуск (simple\_program.py) и удаление (uninstall.sh).

#### **1. Установка (install.sh):**

- Скрипт install.sh подготавливает необходимую среду для работы программы, такие как создание необходимых файлов или создание метки (файл installed.flag), указывающей, что программа установлена.

#### **2. Запуск программы (simple\_program.py):**

- При запуске программа сначала проверяет, была ли она установлена через наличие файла installed.flag. Если файл отсутствует, программа предлагает сначала пройти установку.
- Затем проверяется предел использования, описанный в файле limit\_data.txt, включая количество запусков (starts) и время использования (time). Если предел достигнут

(превышено MAX\_STARTS или TIME\_LIMIT), пользователю выдается уведомление, и программа закрывается.

- Если предел не достигнут, пользователь вводит свое полное имя, и программа проверяет, есть ли оно в файле user\_data.txt. Если нет, оно записывается после шифрования с помощью SHA-256.

- Пользователь может ввести show, чтобы узнать оставшееся количество запусков и время перед завершением программы.

- На конец программа обновляет лимиты и предупреждает пользователя, если лимиты были достигнуты, с предложением купить полную версию или удалить программу.

### 3. Удаление (uninstall.sh):

- Скрипт uninstall.sh удаляет все файлы и метки, включая installed.flag, user\_data.txt и limit\_data.txt, чтобы полностью удалить программу с системы.

Программа ограничивает количество запусков и время использования, чтобы мотивировать пользователя приобрести полную версию, если он хочет продолжить использовать программу.

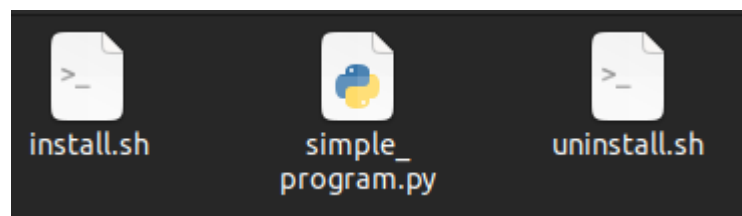


Рисунок 1: Файлы в программе

## 2. 2 Выполнение программы

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ bash install.sh
Файл user_data.txt был создан.
Программа уже была установлена на этом компьютере.
Количество предыдущих запусков: 2, Время использования ранее: 7 секунд.
Программа была успешно установлена. Чтобы запустить программу, используйте команду: './simple_program.py'.
```

Рисунок 2 - Проведение установки программы

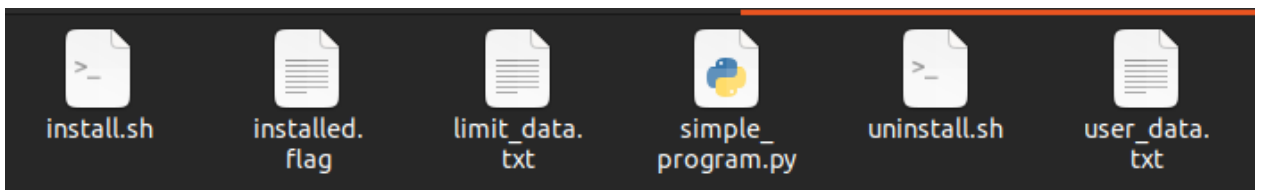


Рисунок 3 - Файлы, создаваемые при успешной установке

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ ./simple_program.py
Введите ваше полное имя: chu
Ваша информация была сохранена.
Введите 'show', чтобы увидеть оставшиеся лимиты, или нажмите Enter, чтобы выйти: show
Осталось запусков: 4, Время: 0 минут 28 секунд.
```

Рисунок 4 – Запуск программы и ввод имени.

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ ./simple_program.py
Введите ваше полное имя: chu
Ваше имя уже сохранено в системе.
Введите 'show', чтобы увидеть оставшиеся лимиты, или нажмите Enter, чтобы выйти: show
Осталось запусков: 2, Время: 0 минут 22 секунд.
```

Рисунок 5 - Тестирование ввода уже введенного имени.

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ ./simple_program.py
Введите ваше полное имя: he
Ваша информация была сохранена.
Введите 'show', чтобы увидеть оставшиеся лимиты, или нажмите Enter, чтобы выйти: show
Осталось запусков: 0, Время: -1 минут 25 секунд.
Достигнут лимит. Пожалуйста, купите полную версию или удалите программу.
```

Рисунок 6 - Уведомление об окончании времени использования.

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ cat user_data.txt
6eff9efbe9ec4716826e179e54e763810cddf48cd0e73c9e6ace6253d377e3d9
372f7e2fd2d01ce2a1d71dc072acbb4c6fd25a1087cd7f153f4ec0ce37e1ede
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ cat limit_data.txt
starts:6
time:118
```

Рисунок 7 - Информация о файлах user\_data.txt и limit\_data.txt

```
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ bash uninstall.sh
Программа была удалена.
Файл user_data.txt был удален.
Файл limit_data.txt сохраняется для сохранения предыдущих данных использования.
```

Рисунок 8 - Проведение удаления приложения.

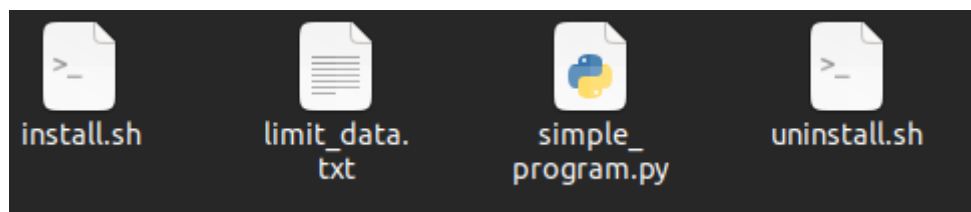


Рисунок 9 - Файлы, остающиеся после удаления приложения.

```

chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ bash install.sh
Файл user_data.txt был создан.
Программа уже была установлена на этом компьютере.
Количество предыдущих запусков: 6, Время использования ранее: 118 секунд.
Программа была успешно установлена. Чтобы запустить программу, используйте команду: './simple_program.py'.
chu@chu-Latitude-5510:~/Documents/Technologies_and_methods_of_programming/Lab_2$ ./simple_program.py
Вы достигли лимита количества запусков. Пожалуйста, купите полную версию или удалите программу.

```

Рисунок 10 - Проведение повторной установки программы.

Мы видим, что из-за окончания лимита использования, даже после повторной установки программы, мы не сможем её использовать.

## 2.2 Код

- File install.sh:

```

#!/bin/bash

INSTALL_DIR="$PWD"
INSTALL_FLAG="$INSTALL_DIR/installed.flag"
LIMIT_FILE="$INSTALL_DIR/limit_data.txt"

# Проверка, была ли программа установлена ранее
if [ -f "$INSTALL_FLAG" ]; then
    echo "Программа уже была установлена ранее. Пожалуйста, удалите ее
перед повторной установкой."
    exit 1
fi

# Создание файла-флага установки
touch "$INSTALL_FLAG"

# Создание файла `user_data.txt`, если он еще не существует
if [ ! -f "$INSTALL_DIR/user_data.txt" ]; then
    touch "$INSTALL_DIR/user_data.txt"
    echo "Файл user_data.txt был создан."
else
    echo "Файл user_data.txt уже существует."
fi

# Проверка, существует ли файл `limit_data.txt` с предыдущей установки
if [ -f "$LIMIT_FILE" ]; then
    echo "Программа уже была установлена на этом компьютере."
    starts=$(grep "starts" "$LIMIT_FILE" | cut -d':' -f2)
    time_used=$(grep "time" "$LIMIT_FILE" | cut -d':' -f2)
    echo "Количество предыдущих запусков: $starts, Время использования
ранее: $time_used секунд."

```

```

else
    # Если файл `limit_data.txt` еще не существует, создаем файл с
    начальным лимитом
    echo "starts:0" > "$LIMIT_FILE"
    echo "time:0" >> "$LIMIT_FILE"
    echo "Файл limit_data.txt был создан."
fi

echo "Программа была успешно установлена. Чтобы запустить программу,
используйте команду: './simple_program.py'."

```

- File simple\_program.py:

```

#!/usr/bin/env python3
import os
import time
import hashlib
import threading
import sys

INSTALL_DIR = os.getcwd()
USER_FILE = os.path.join(INSTALL_DIR, "user_data.txt")
LIMIT_FILE = os.path.join(INSTALL_DIR, "limit_data.txt")
INSTALL_FLAG = os.path.join(INSTALL_DIR, "installed.flag")
MAX_STARTS = 5
TIME_LIMIT = 30 # Ограничение времени использования (в секундах)

def hash_string(s):
    return hashlib.sha256(s.encode()).hexdigest()

def check_user_exists(full_name):
    if os.path.exists(USER_FILE):
        with open(USER_FILE, "r") as file:
            for line in file:
                if hash_string(full_name) in line:
                    return True
    return False

def add_user(full_name):
    with open(USER_FILE, "a") as file:
        file.write(hash_string(full_name) + "\n")

def check_limits():
    """Проверка лимитов доступа и времени."""

```

```

if not os.path.exists(LIMIT_FILE):
    with open(LIMIT_FILE, "w") as file:
        file.write(f"starts:0\ntime:0\n")

with open(LIMIT_FILE, "r") as file:
    data = file.readlines()
    starts = int(data[0].split(":")[1].strip())
    used_time = int(data[1].split(":")[1].strip())

if starts >= MAX_STARTS:
    return False, "Вы достигли лимита количества запусков. Пожалуйста,
купите полную версию или удалите программу."
if used_time >= TIME_LIMIT:
    return False, "Вы достигли лимита времени использования.
Пожалуйста, купите полную версию или удалите программу."

return True, ""

def update_limits(start_time):
    """Обновление количества запусков и времени использования."""
    with open(LIMIT_FILE, "r") as file:
        data = file.readlines()
        starts = int(data[0].split(":")[1].strip()) + 1
        used_time = int(data[1].split(":")[1].strip()) + int(time.time() -
start_time)

    with open(LIMIT_FILE, "w") as file:
        file.write(f"starts:{starts}\ntime:{used_time}\n")

    return starts, used_time

def countdown_timer(remaining_time, program_running):
    """Обратный отсчет времени использования."""
    while remaining_time > 0 and program_running.is_set():
        time.sleep(1)
        remaining_time -= 1
    if remaining_time <= 0:
        program_running.clear()

def check_installation():
    """Проверка, установлена ли программа."""
    if not os.path.exists(INSTALL_FLAG):
        print("Программа не установлена. Пожалуйста, установите ее перед
запуском.")
        sys.exit(1)

```



```

def main():
    check_installation()
    allowed, message = check_limits()
    if not allowed:
        print(message)
        sys.exit(1)

    start_time = time.time()
    program_running = threading.Event()
    program_running.set()
    remaining_time = TIME_LIMIT
    timer_thread = threading.Thread(target=countdown_timer,
args=(remaining_time, program_running))
    timer_thread.start()

    try:
        full_name = input("Введите ваше полное имя: ")
        if not program_running.is_set():
            print("\nВремя использования истекло.")
            sys.exit(1)

        if check_user_exists(full_name):
            print("Ваше имя уже сохранено в системе.")
        else:
            add_user(full_name)
            print("Ваша информация была сохранена.")

        action = input("Введите 'show', чтобы увидеть оставшиеся лимиты,
или нажмите Enter, чтобы выйти: ")
        if action == "show":
            starts, used_time = update_limits(start_time)
            remaining_starts = MAX_STARTS - starts
            remaining_time_overall = TIME_LIMIT - used_time
            remaining_minutes = remaining_time_overall // 60
            remaining_seconds = remaining_time_overall % 60
            print(f"Осталось запусков: {remaining_starts}, Время:
{remaining_minutes} минут {remaining_seconds} секунд.")

            program_running.clear()
        except KeyboardInterrupt:
            print("\nПрограмма была остановлена.")
            program_running.clear()

    timer_thread.join()

```

```

starts, used_time = update_limits(start_time)
if used_time >= TIME_LIMIT or starts >= MAX_STARTS:
    print("Достигнут лимит. Пожалуйста, купите полную версию или
удалите программу.")
    sys.exit(1)

if __name__ == "__main__":
    main()

```

#### - File uninstall.sh

```

#!/bin/bash

INSTALL_DIR="$PWD"
INSTALL_FLAG="$INSTALL_DIR/installed.flag"

# Удаление файла-флага установки
if [ -f "$INSTALL_FLAG" ]; then
    rm "$INSTALL_FLAG"
    echo "Программа была удалена."
else
    echo "Программа не установлена."
    exit 1
fi

# Удаление файла user_data.txt, но сохранение limit_data.txt
if [ -f "$INSTALL_DIR/user_data.txt" ]; then
    rm "$INSTALL_DIR/user_data.txt"
    echo "Файл user_data.txt был удален."
fi

# Сохранение файла limit_data.txt для сохранения предыдущих данных лимита
if [ -f "$INSTALL_DIR/limit_data.txt" ]; then
    echo "Файл limit_data.txt сохраняется для сохранения предыдущих данных
использования."
fi

```

## ЗАКЛЮЧЕНИЕ

Программа включает этапы установки, запуска и удаления, причем важным элементом является ограничение на количество запусков и время использования. Это мероприятие призвано стимулировать пользователей к приобретению полной версии. При необходимости пользователь может легко удалить программу из системы с помощью `uninstall.sh`. Такой подход гарантирует удобство и простоту в использовании программы.