



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ
КАФЕДРА

«Информатика и системы управления» (ИУ)
«Программное обеспечение ЭВМ и информационные технологии» (ИУ7)

ОТЧЁТ ПО УЧЕБНОЙ ПРАКТИКЕ

Тип практики: Проектно-технологическая

Название предприятия: НУК ИУ МГТУ им. Н. Э. Баумана

Студент:

Темирканов Кантемир Асланович, группа ИУ7-13Б

(подпись, дата)

Руководитель от предприятия:

Старший преподаватель ИУ-7

Ломовской Игорь Владимирович

Руководитель от кафедры:

Ассистент кафедры ИУ-7

Кострицкий Александр Сергеевич

**Error! Reference
source not found.**

**Error! Reference
source not found.**

Оценка: _____

Оглавление

| | |
|-----------------------------------|----|
| Постановка задачи | 2 |
| Особенности реализации..... | 3 |
| Ограничения по безопасности | 6 |
| Различные тестовые случаи | 7 |
| Инструкции пользователя | 11 |
| Листинг исходного кода..... | 13 |

Постановка задачи

Разработать скрипт командной оболочки для обработки текстовых файлов.

1. У скрипта есть список расширений *временных файлов*. По умолчанию список состоит из «log».
2. У скрипта есть список расширений *рабочих файлов*. По умолчанию список состоит из «py».
3. У скрипта есть *рабочая папка*, в которой выполняется вся работа скрипта. По умолчанию это папка самого скрипта.
4. Настройки скрипта сохраняются в файле «.myconfig» рядом со скриптом. Если файл при запуске нельзя обнаружить, генерируется файл настроек по умолчанию.
5. У скрипта есть *записанная пользователем* в виде строки команда. По умолчанию это «grep def* program.py > last.log».

Скрипт должен предоставлять пользователю с помощью меню и текстового интерфейса следующие возможности:

1. Возможность просмотреть или задать заново список расширений временных файлов.
2. Возможность добавлять или удалять конкретное расширение из списка расширений временных файлов. Достаточно реализовать удаление по номеру.
3. Возможность просмотреть или задать заново список расширений рабочих файлов.
4. Возможность добавлять или удалять конкретное расширение из списка расширений рабочих файлов. Достаточно реализовать удаление по номеру.
5. Возможность просмотреть, изменить или задать заново рабочую папку скрипта.
6. Возможность удалить временные файлы.
7. Возможность выполнить или изменить записанную команду.
8. Возможность просмотреть все строки, ограниченные апострофами, во всех рабочих файлах.
9. Возможность просмотреть объем каждого временного файла.

Скрипт должен иметь возможность запуска в тихом режиме (без меню), для чего следует использовать позиционные аргументы. Скрипт не должен позволять запуск от имени администратора.

Особенности реализации

В начале скрипта происходит объявление функций, которые выполняют все операции скрипта.

Список функций и их назначение:

| Название функции | Назначение функции |
|---------------------|---|
| pr_list | Построчный вывод списка расширений |
| make_list | Создание списка расширений |
| append_list | Добавление в конец списка нового расширения |
| del_list | Удаление расширения из списка по номеру |
| make_folder | Выбор рабочей папки |
| del_temp | Удаление всех временных файлов |
| make_command | Запись команды |
| pr_strings | Поиск строк, ограниченных апострофами, во всех рабочих файлах |
| read_config | Чтение файла настроек |
| pr_sizes | Вывод размеров всех временных файлов |
| make_temp_silent | Создание списков расширений временных файлов в тихом режиме |
| make_work_silent | Создание списков расширений рабочих файлов в тихом режиме |
| append_silent | Добавление в конец списка нового расширения в тихом режиме |
| del_silent | Удаление расширения из списка по номеру в тихом режиме |
| make_folder_silent | Выбор рабочей папки в тихом режиме |
| make_command_silent | Запись команды в тихом режиме |
| make_config | Создание файла настроек |
| write_config | Запись в файл настроек |

Далее происходит работа с файлом “.myconfig”. Скрипт проходит по папке, в которой находится, и ищет файл с данным названием. Если такой файл найти не удалось, то создается новый файл, в который записываются переменные по умолчанию.

Пример создания файла по умолчанию :

```
config="$(dirname "$0")/.myconfig"
if [ -f "$config" ] ; then
    read_config
else
    make_config
    read_config
fi
```

Если файл успешно найден, то происходит присвоение каждой строки файла конкретной переменной.

Пример файла “.myconfig” по умолчанию :

```
*.log
*.py
grep def* program.py >last.log
/users/project/
```

Далее идет проверка запуска в интерактивном режиме.

Пример запуска интерактивного режима:

```
if [ "$1" != -s ] ; then
    #Интерактивный режим
else
    #Тихий режим
fi
```

Интерактивный режим реализуется с помощью бесконечного цикла, из которого пользователь может выйти отдельной командой.

Пример запуска интерактивного меню и его вывод:

```
menu_number=1
while [ "$menu_number" != "0" ] ; do
    echo "1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих файлах
16. Просмотреть объём каждого временного файла
```

```
0. Завершить скрипт"
    echo -n "Выберите пункт меню (от 0 до 14): "
    read -r menu_number
```

Тихий режим реализуется с помощью позиционных параметров, передаваемых пользователем. Он запускается в том случае, если первый параметр “-s”.

В случае использования команд, которые вызывают изменение глобальных переменных, происходит перезапись файла конфига с помощью функции `write_config`.

После выполнения всех функций в тихом режиме скрипт завершается командой `exit`.

Ограничения по безопасности

В сценарии предусмотрена проверка на запуск от имени суперпользователя. Пользователь при желании может прекратить работу.

Пример обработки запуска от суперпользователя:

```
if [ $EUID -eq 0 ]; then
    echo -n "Вы запустили программу от суперпользователя, вы уверены что
хотите продолжить? [д/Н]: "
    read -r userPrikolist;
    if [ "$userPrikolist" != "д" ]; then
        exit 1
    fi
fi
```

Невалидные запросы, при которых поведение скрипта не определено :

1. Записанная пользовательская команда содержит ошибки, имеет недопустимое выполнение или нехватку прав.
2. Путь к рабочей директории недоступен к чтению, указан неверно или содержит в себе пробелы и указан без кавычек.
3. Если существует некорректный файл настроек “.myconfig” в папке скрипта.
4. Скрипт был вызван по относительному пути.
5. Рабочая папка с пробелами в пути задана без кавычек

Различные тестовые случаи

Пример изменения команды и ее выполнение в рабочей директории :

```
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 14
Введите команду: ls -la
./myconfig
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 13
ls -la
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
```



```

12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 12
итого 444
drwxr-xr-x 2 kan kan 4096 дек 27 19:58 .
drwxr-xr-x 25 kan kan 4096 дек 22 15:13 ..
-rw-rw-r- 1 kan kan 11055 дек 27 19:58 itog.sh
-rw-rw-r- 1 kan kan 20 дек 27 19:59 .myconfig
-rw-rw-r- 1 kan kan 126152 дек 22 15:11 Отчет.docx
-rw-rw-r- 1 kan kan 299857 дек 22 15:10 Отчет.pdf

```

Пример изменения директория и выполнения 15 команды :

```

1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 10
Введите новую папку: folder
./myconfig
1. Просмотреть
список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла

```

```
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 15
'Hello, world!'
'It\'s Linux project!'
'Big Value'
```

Пример добавления расширения и исполнения 16 команды :

```
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 3
Введите расширение в виде '*.расширение': *.txt
./myconfig
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 16
4,0K ./2.txt
```

Пример вывода ошибки при удалении по индексу :

```
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
```

```

3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 4
Введите номер удаляемого расширения: 0
Неверный номер.
./myconfig
1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку
скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт
Выберите пункт меню (от 0 до 14): 4
Введите номер удаляемого расширения: 1

```

Инструкции пользователя

Описание

Обработка файлов в каталоге по расширениям с помощью параметров в тихом режиме или интерактивного меню.

По умолчанию, если нет параметров, вызывается интерактивный режим, в противном случае тихий.

Тихий режим

Параметры :

| | |
|----------|--|
| prwrk | Вывод расширений рабочих файлов |
| prtmp | Вывод расширений временных файлов |
| mktmp | Создание нового списка временных файлов |
| mkwrk | Создание нового списка рабочих файлов |
| addtmp | Добавление расширений в список временных файлов |
| addwrk | Добавление расширений в список рабочих файлов |
| deltmp | Удаление расширений из списка временных файлов по номеру |
| delwrk | Удаление расширений из списка рабочих файлов по номеру |
| prfolder | Вывод рабочей папки |
| mkfolder | Выбор рабочей папки |
| delfiles | Удаление временных файлов |
| execcmd | Выполнение команд записанных пользователем |
| prcmd | Вывод записанной команды |
| mkcmd | Запись команды |
| prstr | Вывод строк, ограниченных апострофами, из рабочих файлов |
| prsizes | Вывод размеров временных файлов |

Интерактивный режим

Меню с описанием команд:

1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт

На вход принимается соответствующий номер команды [0-16]. В некоторых пунктах (2,3,4,6,7,8,10,14) потребуется дополнительный ввод, например, требуемое значение.

Листинг исходного кода

```
#!/bin/bash

if [ $EUID -eq 0 ]; then
    echo -n "Вы запустили программу от суперпользователя, вы уверены что  
хотите продолжить? [д/Н]: "
    read -r userPrikolist;
    if [ "$userPrikolist" != "д" ]; then
        exit 1
    fi
fi

pr_list() {
    if [ "$1" == "work" ] ; then
        echo "Список расширений рабочих файлов: "
        list="$work_file"
    elif [ "$1" == "temp" ] ; then
        echo "Список расширений временных файлов: "
        list="$temp_file"
    fi

    set -f
    for i in $list ; do
        echo "$i"
    done
    set +f
}

make_list() {
    if [ "$1" == "work" ] ; then
        echo -n "Введите количество расширений рабочих файлов: "
        read -r amount
    elif [ "$1" == "temp" ] ; then
        echo -n "Введите количество расширений временных файлов: "
        read -r amount
    fi
    list=""
    set -f
    while [ "$amount" -gt 0 ] ; do
        echo -n "Введите расширение в виде '*.расширение': "
        read -r extension
        if echo "$extension" | grep -Eq "^\*\.[A-Za-z0-9]+$" ; then
            list+=" $extension"
            amount=$((amount - 1))
        else
            echo "Неверный формат расширения."
        fi
    done
    set +f
    if [ "$1" == "work" ] ; then
        work_file="$list"
    elif [ "$1" == "temp" ] ; then
        temp_file="$list"
    fi
    write_config
}
```

```

append_list() {
    set -f
    echo -n "Введите расширение в виде '*.расширение': "
    read -r extension
    if echo "$extension" | grep -Eq "^\*[A-Za-z0-9]+$" ; then
        if [ "$1" == "work" ] ; then
            work_file+=" $extension"
        elif [ "$1" == "temp" ] ; then
            temp_file+=" $extension"
        fi
    else
        echo "Неверный формат расширения."
    fi
    set +f
    write_config
}

del_list() {
    set -f
    echo -n "Введите номер удаляемого расширения: "
    read -r number
    if echo "$number" | grep -Eq "^[1-9][0-9]*$" ; then
        counter=1
        list=''
        if [ "$1" == "work" ] ; then
            for i in $work_file ; do
                if [ "$counter" -ne "$number" ] ; then
                    list+=" $i"
                fi
                counter=$((counter + 1))
            done
            work_file="$list"
        elif [ "$1" == "temp" ] ; then
            for i in $temp_file ; do
                if [ "$counter" -ne "$number" ] ; then
                    list+=" $i"
                fi
                counter=$((counter + 1))
            done
            temp_file="$list"
        fi
    else
        echo "Неверный номер."
    fi
    set +f
    write_config
}

make_folder() {
    echo -n "Введите новую папку: "
    read -r folder
    IFS_old="$IFS"
    IFS=' '
    if [ -d "$folder" ] ; then
        work_folder="$folder"
    fi
}

```

```

        cd "$folder"
    else
        echo "Несуществующая папка."
    fi
    IFS="$IFS_old"
    write_config
}

del_temp() {
    for i in $temp_file ; do
        file="./$i"
        if [ -f "$file" ] ; then
            rm -f "$file"
        fi
    done
}

make_command() {
    echo -n " Введите команду: "
    read -r command
    if [ -n "$command" ] ; then
        written_command="$command"
        write_config
    else
        echo "Пустая строка."
    fi
}

pr_strings() {
    for i in $work_file ; do
        file="./$i"
        echo " $file:"
        if [ -f "$file" ] ; then
            grep -Eo "'.*'" "$file"
        fi
    done
}

pr_sizes() {
    for i in $temp_file ; do
        file="./$i"
        if [ -f "$file" ] ; then
            du -h "$file"
        fi
    done
}

make_temp_silent() {
    set -f
    list=''
    for i in "$@" ; do
        if echo "$i" | grep -Eq "^\*\. [A-Za-z0-9]+$" ; then
            list+=" $i"
        fi
    done
}

```



```

    if [ -n "$list" ] ; then
        temp_file="$list"
        write_config
    fi
    set +f
}

make_work_silent() {
    set -f
    list=''
    for i in "$@" ; do
        if echo "$i" | grep -Eq "^\*[A-Za-z0-9]+$" ; then
            list+=" $i"
        fi
    done
    if [ -n "$list" ] ; then
        work_file="$list"
        write_config
    fi
    set +f
}

append_silent() {
    if echo "$1" | grep -Eq "^\*[A-Za-z0-9]+$" ; then
        if [ "$2" == "work" ] ; then
            work_file+=" $extension"
        elif [ "$2" == "temp" ] ; then
            temp_file+=" $extension"
        fi
    fi
    set +f
    write_config
}

del_silent() {
    if echo "$1" | grep -Eq "^[1-9][0-9]*$" ; then
        counter=1
        list=''
        if [ "$2" == "work" ] ; then
            for i in $work_file ; do
                if [ "$counter" -ne "$number" ] ; then
                    list+=" $i"
                fi
                counter=$((counter + 1))
            done
            work_file="$list"
        elif [ "$2" == "temp" ] ; then
            for i in $temp_file ; do
                if [ "$counter" -ne "$number" ] ; then
                    list+=" $i"
                fi
                counter=$((counter + 1))
            done
            temp_file="$list"
        fi
    fi
    write_config
}

```

```

}

make_folder_silent() {
    IFS_old="$IFS"
    IFS=""
    if [ -d "$1" ] ; then
        work_folder="$1"
    fi
    write_config

    cd "$work_folder"
    pwd
    IFS="$IFS_old"
}

make_command_silent() {
    if [ -n "$1" ] ; then
        written_command="$1"
        write_config
    fi
}

make_config() {
    config="$(dirname "$0")/.myconfig"
    echo "/*.log" >> "$config"
    echo "/*.py" >> "$config"
    dirname "$0" >> "$config"
    echo "grep def* program.py >last.log" >> "$config"
}

write_config() {
    config="$(dirname "$0")/.myconfig"
    echo "$config"
    echo "$temp_file" > "$config"
    echo "$work_file" >> "$config"
    echo "$work_folder" >> "$config"
    echo "$written_command" >> "$config"
}

read_config() {
    config="$(dirname "$0")/.myconfig"
    #echo "$config"
    temp_file=$(head -n 1 "$config" | tail -n 1)
    work_file=$(head -n 2 "$config" | tail -n 1)
    work_folder=$(head -n 3 "$config" | tail -n 1)
    IFS_old="$IFS"
    IFS=""
    cd "$work_folder"
    IFS="$IFS_old"
    written_command=$(head -n 4 "$config" | tail -n 1)
}

```

```

config="$(dirname "$0")/.myconfig"
if [ -f "$config" ] ; then
    read_config
else
    make_config
    read_config
fi

if [ "$1" != -s ] ; then
    menu_number=1
    while [ "$menu_number" != "0" ] ; do
        echo "1. Просмотреть список расширений временных файлов
2. Задать заново список расширений временных файлов
3. Добавить расширение в список расширений временных файлов
4. Удалить расширение из списка расширений временных файлов
5. Просмотреть список расширений рабочих файлов
6. Задать заново список расширений рабочих файлов
7. Добавить элемент в список расширений рабочих файлов
8. Удалить элемент из списка расширений рабочих файлов
9. Просмотреть рабочую папку скрипта
10. Задать заново рабочую папку скрипта
11. Удалить временные файлы
12. Выполнить записанную команду
13. Просмотреть записанную команду
14. Изменить записанную команду
15. Просмотреть все строки, ограниченные апострофами во всех рабочих
файлах
16. Просмотреть объём каждого временного файла
0. Завершить скрипт"
        echo -n "Выберите пункт меню (от 0 до 14): "
        read -r menu_number

        if [ "$menu_number" == "1" ] ; then
            pr_list temp
        elif [ "$menu_number" == "2" ] ; then
            make_list temp
        elif [ "$menu_number" == "3" ] ; then
            append_list temp
        elif [ "$menu_number" == "4" ] ; then
            del_list temp
        elif [ "$menu_number" == "5" ] ; then
            pr_list work
        elif [ "$menu_number" == "6" ] ; then
            make_list work
        elif [ "$menu_number" == "7" ] ; then
            append_list work
        elif [ "$menu_number" == "8" ] ; then
            del_list work
        elif [ "$menu_number" == "9" ] ; then
            echo "$work_folder"
        elif [ "$menu_number" == "10" ] ; then
            make_folder
        elif [ "$menu_number" == "11" ] ; then
            del_temp
        elif [ "$menu_number" == "12" ] ; then
            eval "$written_command"
        elif [ "$menu_number" == "13" ] ; then
            echo "$written_command"

```

```

        elif [ "$menu_number" == "14" ] ; then
            make_command
        elif [ "$menu_number" == "15" ] ; then
            pr_strings
        elif [ "$menu_number" == "16" ] ; then
            pr_sizes
        fi
    done
else
    if [ "$2" == "prwrk" ] ; then
        pr_list work

    elif [ "$2" == "prtmp" ] ; then
        pr_list temp

    elif [ "$2" == "mktmp" ] ; then
        shift
        shift
        extensions=''
        for i in "$@" ; do
            extensions+=" $i"
        done
        make_temp_silent $extensions

    elif [ "$2" == "mkwrk" ] ; then
        shift
        shift
        extensions=''
        for i in "$@" ; do
            extensions+=" $i"
        done
        make_work_silent $extensions

    elif [ "$2" == "addtmp" ] ; then
        append_silent "$3" temp

    elif [ "$2" == "addwrk" ] ; then
        append_silent "$3" work

    elif [ "$2" == "deltmp" ] ; then
        del_silent "$3" temp

    elif [ "$2" == "delwrk" ] ; then
        del_silent "$3" work

    elif [ "$2" == "prfolder" ] ; then
        echo "$work_folder"
        pwd

    elif [ "$2" == "mkfolder" ] ; then
        shift
        shift
        dir=''
        for i in "$@" ; do
            dir+=" $i"
        done
        dir="{dir:1}"
        make_folder_silent "$dir"

```

```
elif [ "$2" == "delfiles" ] ; then
    del_temp

elif [ "$2" == "execcmd" ] ; then
    eval "$written_command"

elif [ "$2" == "prcmd" ] ; then
    echo "$written_command"

elif [ "$2" == "mkcmd" ] ; then
    make_command_silent "$3"

elif [ "$2" == "prstr" ] ; then
    pr_strings

elif [ "$2" == "prsizes" ] ; then
    pr_sizes
fi
fi
exit
```