|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 | «Информатика и системы управления» (ИУ) |
| Кафедра | «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: Постнов Степан Андреевич, группа ИУ7-11Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |
| Руководитель от предприятия: Старший преподаватель ИУ-7  Ломовской Игорь Владимирович | (подпись, дата) |
| Руководитель от кафедры: Ассистент кафедры ИУ-7  Кострицкий Александр Сергеевич | (подпись, дата) |

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Оглавление**

[Постановка задачи 3](#_Toc91016412)

[Особенности реализации 4](#_Toc91016413)

[Ограничения по безопасности 7](#_Toc91016414)

[Различные тестовые случаи 9](#_Toc91016415)

[Инструкция пользователя 17](#_Toc91016416)

[Листинг исходного кода 20](#_Toc91016417)

# **Постановка задачи**

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

1. Возможность просмотреть или задать заново список *временных* файлов.
2. Возможность добавлять или удалять конкретное расширение из списка *временных* файлов.
3. Возможность просмотреть или задать заново список *рабочих* файлов.
4. Возможность добавлять или удалять конкретное расширение из списка *рабочих* файлов.
5. Возможность просмотреть или задать заново *рабочую папку* скрипта.
6. Возможность удалить *временные* файлы.
7. Возможность выполнить или изменить записанную команду.
8. Возможность просмотреть число строк и слов в каждом *рабочем* файле.
9. Возможность просмотреть объем каждого мусорного файла.

Скрипт должен иметь возможность запуска в *тихом* режиме (без меню), для чего следует использовать позиционные аргументы.

# **Особенности реализации**

Скрипт, под названием «*work.sh*» в начале работы создает скрытый файл «*.myconfig*», в котором содержатся настройки скрипта построчно:

* Список расширений временных файлов (по умолчанию «*log*»)
* Список расширений рабочих файлов (по умолчанию «*txt*»)
* Путь к рабочей папке (по умолчанию *папка скрипта*)
* Записанную команду (по умолчанию «*grep error\* last.txt > last.log*»)

В случае отсутствия данного файла, при запуске скрипта она создастся автоматически с настройками по умолчанию.

Листинг 1. Создание скрытого файла с настройками скрипта, если его нет

1. **if** [ ! -e "$script\_folder"".myconfig" ]; **then**
2. echo "log " > "$script\_folder"".myconfig"
3. {
4. echo "txt "
5. echo "$script\_folder"
6. echo "grep error\* last.txt > last.log"
7. } >> "$script\_folder"".myconfig"
8. **fi**

Пользователь может взаимодействовать со скриптом, запущенным в *интерактивном* режиме, с помощью меню. Выбрав пункт от 0 до 10, пользователь запустит одну из команд меню. При вводе некорректного значения номера пункта, скрипт поведет себя штатно и выведет меню повторно.

1. Задать заново список временных файлов
2. Добавить или удалить конкретное расширение из списка временных файлов
3. Задать заново список рабочих файлов
4. Добавить или удалить конкретное расширение из списка рабочих файлов
5. Просмотреть или задать заново рабочую папку скрипта
6. Удалить временные файлы
7. Выполнить или изменить записанную команду
8. Просмотреть число строк и слов в каждом рабочем файле
9. Просмотреть объем каждого мусорного файла
10. Выход из меню

Пример 1. Вид меню в интерактивном режиме

Некоторые пункты меню содержат подпункты, которые также требуют ввести необходимое числовое значение для продолжения выполнения команды. Если пользователь введет некорректное значение, скрипт, как и в случае ввода некорректного значения при выборе пункта самого меню, поведет себя штатно и заново выведет меню для дальнейшей работы.

1. Введите желаемый номер меню: 2
2. Выберите необходимый пункт:
3. 1 -- Добавить конкретное расширение
4. 2 -- Удалить конкретное расширение (по значению)
5. :

Пример 2. Выбор подпункта меню

Скрипт также может быть запущен в *тихом* режиме. Для этого стоит передать не менее одного позиционного аргумента. После изменения каких-либо расширений, записанной команды или *рабочей папки*, будет перезаписан файл конфигурации скрипта «*.myconfig*», содержащий настройки по умолчанию

Листинг 2. Выполнение команды в тихом режиме

1. **elif** [[ $1 == "do" ]]; **then** #Исполнение записанной команды скрипта
2. eval "$cur\_command"

# **Ограничения по безопасности**

В скрипте реализованы следующие проверки на некорректный ввод:

1. При некорректном вводе пункта меню произойдет вывод текстового меню заново для дальнейшей работы.
2. При некорректном вводе подпункта к любому пункту меню произойдет вывод текстового меню заново для дальнейшей работы.
3. При попытке запуска скрипта от имени *суперпользователя* выведется предупреждение, которое будет обрывать скрипт при определенном варианте ответа пользователя или же продолжать работу при ином ответе.
4. **if** [ $EUID -**eq** 0 ]; **then**
5. read -rp "Вы запустили скрипт от имени суперпользователя. Вы уверены, что хотите продолжить?[Y/n]: " Y\_n
6. **if** [[ $Y\_n == "n" ]]; **then**
7. exit
8. **fi**
9. **fi**

Листинг 3. Проверка на суперпользователя

Условия, которые должны быть выполнены для работы скрипта:

1. Расширения *временных* и *рабочих* файлов должны быть заданы буквами или цифрами
2. Путь к новой *рабочей папке* должен вести к существующему каталогу, доступному для чтения и выполнения. Также пользователь должен ввести путь ненулевой длины. Путь может быть как относительным, так и абсолютным.
3. Записанная команда скрипта не должна содержать ошибок, нехватку прав. Нежелательные команды, такие как «*rm -rf*», могут привести к неопределенности в работе скрипта.

# **Различные тестовые случаи**

1. stepan@stepan-VirtualBox:~$ ./work.sh
3. =======================================
4. Список расширений временных файлов:
6. 1 расширение - \*.log
8. Список расширений рабочих файлов:
10. 1 расширение - \*.txt
12. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

15. Текущая рабочая папка скрипта: /home/stepan/
17. 1. Задать заново список временных файлов
18. 2. Добавить или удалить конкретное расширение из списка временных файлов
19. 3. Задать заново список рабочих файлов
20. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
21. 5. Просмотреть или задать заново рабочую папку скрипта
22. 6. Удалить временные файлы
23. 7. Выполнить или изменить записанную команду
24. 8. Просмотреть число строк и слов в каждом рабочем файле
25. 9. Просмотреть объем каждого мусорного файла
26. 0. Выход из меню
28. Введите желаемый номер меню: 7
29. Выберите необходимый пункт:
30. 1 -- Выполнить записанную команду
31. 2 -- Изменить записанную команду
32. : 2
33. Введите новую команду: wc -c work.sh
35. \_\_\_\_\_\_\_Команда изменена\_\_\_\_\_\_\_

Пример 3. Изменение записанной команды скрипта и демонстрация работы

3. =======================================
4. Список расширений временных файлов:
6. 1 расширение - \*.log
8. Список расширений рабочих файлов:
10. 1 расширение - \*.txt
12. Записанная команда скрипта: <<wc -c work.sh>>

15. Текущая рабочая папка скрипта: /home/stepan/
17. 1. Задать заново список временных файлов
18. 2. Добавить или удалить конкретное расширение из списка временных файлов
19. 3. Задать заново список рабочих файлов
20. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
21. 5. Просмотреть или задать заново рабочую папку скрипта
22. 6. Удалить временные файлы
23. 7. Выполнить или изменить записанную команду
24. 8. Просмотреть число строк и слов в каждом рабочем файле
25. 9. Просмотреть объем каждого мусорного файла
26. 0. Выход из меню
28. Введите желаемый номер меню: 7
29. Выберите необходимый пункт:
30. 1 -- Выполнить записанную команду
31. 2 -- Изменить записанную команду
32. : 1
33. 14206 work.sh
35. \_\_\_\_\_\_\_Команда выполнена\_\_\_\_\_\_\_

2. =======================================
3. Список расширений временных файлов:
5. 1 расширение - \*.log
7. Список расширений рабочих файлов:
9. 1 расширение - \*.txt
11. Записанная команда скрипта: <<wc -c work.sh>>

14. Текущая рабочая папка скрипта: /home/stepan/
16. 1. Задать заново список временных файлов
17. 2. Добавить или удалить конкретное расширение из списка временных файлов
18. 3. Задать заново список рабочих файлов
19. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
20. 5. Просмотреть или задать заново рабочую папку скрипта
21. 6. Удалить временные файлы
22. 7. Выполнить или изменить записанную команду
23. 8. Просмотреть число строк и слов в каждом рабочем файле
24. 9. Просмотреть объем каждого мусорного файла
25. 0. Выход из меню
27. Введите желаемый номер меню:
28. stepan@stepan-VirtualBox:~$ ./work.sh
30. =======================================
31. Список расширений временных файлов:
33. 1 расширение - \*.log

Пример 4. Изменение рабочей папки скрипта и последующий вывод числа строк и слов во всех рабочих файлах в рабочей папке

1. Список расширений рабочих файлов:
3. 1 расширение - \*.txt
5. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

8. Текущая рабочая папка скрипта: /home/stepan/
10. 1. Задать заново список временных файлов
11. 2. Добавить или удалить конкретное расширение из списка временных файлов
12. 3. Задать заново список рабочих файлов
13. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
14. 5. Просмотреть или задать заново рабочую папку скрипта
15. 6. Удалить временные файлы
16. 7. Выполнить или изменить записанную команду
17. 8. Просмотреть число строк и слов в каждом рабочем файле
18. 9. Просмотреть объем каждого мусорного файла
19. 0. Выход из меню
21. Введите желаемый номер меню: 5
22. Выберите необходимый пункт:
23. 1 -- Просмотреть рабочую папку
24. 2 -- Задать заново рабочую папку
25. : 2
26. Введите путь к новой рабочей папке: ~/
28. \_\_\_\_\_\_\_Рабочая папка задана\_\_\_\_\_\_\_

31. =======================================
32. Список расширений временных файлов:
34. 1 расширение - \*.log
36. Список расширений рабочих файлов:

1. 1 расширение - \*.txt
3. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

6. Текущая рабочая папка скрипта: /home/stepan/
8. 1. Задать заново список временных файлов
9. 2. Добавить или удалить конкретное расширение из списка временных файлов
10. 3. Задать заново список рабочих файлов
11. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
12. 5. Просмотреть или задать заново рабочую папку скрипта
13. 6. Удалить временные файлы
14. 7. Выполнить или изменить записанную команду
15. 8. Просмотреть число строк и слов в каждом рабочем файле
16. 9. Просмотреть объем каждого мусорного файла
17. 0. Выход из меню
19. Введите желаемый номер меню: 8
20. В файле /home/stepan/answer.txt: 98 строк и 354 слов
21. В файле /home/stepan/boom.txt: 0 строк и 0 слов
22. В файле /home/stepan/linux\_ind\_zad.txt: 20 строк и 54 слов
23. В файле /home/stepan/qqq.txt: 4 строк и 4 слов
24. В файле /home/stepan/test.txt: 72 строк и 207 слов
26. =======================================
27. Список расширений временных файлов:
29. 1 расширение - \*.log
31. Список расширений рабочих файлов:
33. 1 расширение - \*.txt
35. Записанная команда скрипта: <<grep error\* last.txt > last.log>>
36. stepan@stepan-VirtualBox:~$ ./work.sh
38. =======================================
39. Список расширений временных файлов:
41. 1 расширение - \*.log
43. Список расширений рабочих файлов:
45. 1 расширение - \*.txt
47. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

50. Текущая рабочая папка скрипта: /home/stepan/


54. Текущая рабочая папка скрипта: /home/stepan/
56. 1. Задать заново список временных файлов
57. 2. Добавить или удалить конкретное расширение из списка временных файлов
58. 3. Задать заново список рабочих файлов
59. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
60. 5. Просмотреть или задать заново рабочую папку скрипта
61. 6. Удалить временные файлы
62. 7. Выполнить или изменить записанную команду
63. 8. Просмотреть число строк и слов в каждом рабочем файле
64. 9. Просмотреть объем каждого мусорного файла
65. 0. Выход из меню
67. Введите желаемый номер меню:

Пример 5. Вывод объема всех мусорных файлов

1. stepan@stepan-VirtualBox:~$ ./work.sh
3. =======================================
4. Список расширений временных файлов:
6. 1 расширение - \*.log
8. Список расширений рабочих файлов:
10. 1 расширение - \*.txt
12. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

15. Текущая рабочая папка скрипта: /home/stepan/
17. 1. Задать заново список временных файлов
18. 2. Добавить или удалить конкретное расширение из списка временных файлов
19. 3. Задать заново список рабочих файлов
20. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
21. 5. Просмотреть или задать заново рабочую папку скрипта
22. 6. Удалить временные файлы
23. 7. Выполнить или изменить записанную команду
24. 8. Просмотреть число строк и слов в каждом рабочем файле
25. 9. Просмотреть объем каждого мусорного файла
26. 0. Выход из меню
28. Введите желаемый номер меню: 9
29. Файл /home/stepan/bruh.log имеет размер: 0 байтов
30. Файл /home/stepan/okey.log имеет размер: 1005 байтов
31. Файл /home/stepan/omg.log имеет размер: 0 байтов
32. Файл /home/stepan/privet.log имеет размер: 328 байтов
33. Файл /home/stepan/test.log имеет размер: 0 байтов

Пример 6. Пользователь задает новый список расширений временных файлов

1. 1. Задать заново список временных файлов
2. 2. Добавить или удалить конкретное расширение из списка временных файлов
3. 3. Задать заново список рабочих файлов
4. 4. Добавить или удалить конкретное расширение из списка рабочих файлов
5. 5. Просмотреть или задать заново рабочую папку скрипта
6. 6. Удалить временные файлы
7. 7. Выполнить или изменить записанную команду
8. 8. Просмотреть число строк и слов в каждом рабочем файле
9. 9. Просмотреть объем каждого мусорного файла
10. 0. Выход из меню
12. Введите желаемый номер меню: 1
13. Введите название нового расширения файлов: jpeg
14. Введите название нового списка расширений или нажмите "Enter" для выхода: mp3
15. Введите название нового списка расширений или нажмите "Enter" для выхода:
17. \_\_\_\_\_\_\_Список задан заново\_\_\_\_\_\_\_

20. =======================================
21. Список расширений временных файлов:
23. 1 расширение - \*.jpeg
24. 2 расширение - \*.mp3
26. Список расширений рабочих файлов:
28. 1 расширение - \*.txt
30. Записанная команда скрипта: <<grep error\* last.txt > last.log>>

33. Текущая рабочая папка скрипта: /home/stepan/

# **Инструкция пользователя**

**КРАТКИЙ ОБЗОР**

work.sh [ПАРАМЕТРЫ] […]

**ОПИСАНИЕ**

Действия с *рабочими* и *временными* файлами с помощью *интерактивного* и *тихого* режимов скрипт. Если при запуске не будут переданы параметры, то скрипт запустится в *интерактивном* режиме. Иначе в *тихом*.

**Тихий режим**

Параметры :

*show* – вывести на экран (всего 2 параметра)

*temp* – список расширений *временных* файлов

*work* – список расширений *рабочих* файлов

*folder* – *рабочую папку*

*sw –* кол-ва строк и слов во всех *рабочих* файлах

*trash –* размера всех мусорных файлов

*add –* добавить к (всего 3 параметра)

*temp –* списку расширений *временных* файлов

*[параметр]* – данное расширение

*work –* списку расширений *рабочих* файлов

*[параметр]* – данное расширение

*del –* удалить (всего от 2 до 3 параметров)

*temp* – из списка *временных* файлов

*-a* – все *временные* файлы

*[параметр]* – данное расширение

*work* – из списка *рабочих* файлов

*[параметр]* – данное расширение

*config* – файл с настройками скрипта «*.myconfig*»

*reload* – задать заново (всего мин. 3 параметра)

*temp* – список расширений *временных* файлов

*[параметры]* – расширения для добавления

*work* – список расширений *рабочих* файлов

*[параметры]* – расширения для добавления

*folder* – путь к *рабочей папке* скрипта

*[параметр]* – путь к новой *рабочей папке*

*command* – команду скрипта

*[параметры]* – составляющие команды

*do* – исполнить записанную команду (всего 1 параметр)

**Интерактивный режим**

Текстовое меню:

1. Задать заново список временных файлов

2. Добавить или удалить конкретное расширение из списка временных файлов

3. Задать заново список рабочих файлов

4. Добавить или удалить конкретное расширение из списка рабочих файлов

5. Просмотреть или задать заново рабочую папку скрипта

6. Удалить временные файлы

7. Выполнить или изменить записанную команду

8. Просмотреть число строк и слов в каждом рабочем файле

9. Просмотреть объем каждого мусорного файла

0. Выход из меню

# **Листинг исходного кода**

|  |
| --- |
| 1. #!/bin/bash 3. #Stepan Postnov ICS7-11B 5. script\_folder="$(realpath "$0" | sed 's/work.sh//g')" 6. #Предупреждение пользователя при запуске скрипта от имени суперпользователя 7. **if** [ $EUID -**eq** 0 ]; **then** 8. read -rp "Вы запустили скрипт от имени суперпользователя. Вы уверены, что хотите продолжить?[Y/n]: " Y\_n 9. **if** [[ $Y\_n == "n" ]]; **then** 10. exit 1 11. **fi** 12. **fi** 13. #Создание скрытого файла .myconfig, если его нет в папке скрипта 14. **if** [ ! -e "$script\_folder"".myconfig" ]; **then** 15. echo "log " > "$script\_folder"".myconfig" 16. { 17. echo "txt " 18. echo "$script\_folder" 19. echo "grep error\* last.txt > last.log" 20. } >> "$script\_folder"".myconfig" 21. **fi** 22. #Объявление переменных для упрощения работы скрипта 23. ext\_temp=$(head -1 "$script\_folder"".myconfig") 24. ext\_work=$(head -2 "$script\_folder"".myconfig" | tail -1) 25. work\_folder=$(tail -2 "$script\_folder"".myconfig" | head -1) 26. cur\_command=$(tail -1 "$script\_folder"".myconfig") 27. #Объявление функции, которая перезаписывает файл скрипта после изменения параметров 28. **function** change\_myconfig { 29. echo "$1" > "$script\_folder"".myconfig" 30. { 31. echo "$2" 32. echo "$3" 33. echo "$4" 34. } >> "$script\_folder"".myconfig" 35. } 36. #\_\_\_\_\_\_\_Тихий режим\_\_\_\_\_\_\_ 37. **if** [[ $1 == "show" ]]; **then** 38. **if** [ "$2" == "temp" ]; **then** #Вывод списка расширений временных файлов 39. sch=0 40. **for** i **in** $ext\_temp; **do** 41. sch=$((sch+1)) 42. echo $sch" расширение - \*.""$i" 43. **done** 44. **elif** [ "$2" == "work" ]; **then** #Вывод списка расширений рабочих файлов 45. sch=0 46. **for** i **in** $ext\_work; **do** 47. sch=$((sch+1)) 48. echo $sch" расширение - \*.""$i" 49. **done** 50. **elif** [ "$2" == "folder" ]; **then** #Вывод содержимого рабочей папки 51. echo -e "Содержимое рабочей папки: \n""$(ls "$work\_folder")" 52. **elif** [ "$2" == "sw" ]; **then** #Вывод кол-ва строк и слов во всех рабочих файлах 53. **for** i **in** $ext\_work; **do** 54. str=$(ls "$work\_folder" | grep \."$i") 55. **if** [[ $str =~ .+ ]]; **then** 56. **for** j **in** "$work\_folder"\*."$i"; **do** 57. echo "В файле $j: $(wc -l < "$j") строк и $(wc -w < "$j") слов" 58. **done** 59. **fi** 60. **done** 61. **elif** [ "$2" == "trash" ]; **then** #Вывод размера всех мусорных файлов 62. **for** i **in** $ext\_temp; **do** 63. str=$(ls "$work\_folder" | grep \."$i") 64. **if** [[ $str =~ .+ ]]; **then** 65. **for** j **in** "$work\_folder"\*."$i"; **do** 66. echo "Файл $j имеет размер: $(wc -c < "$j") байтов" 67. **done** 68. **fi** 69. **done** 70. **fi** 71. **elif** [[ $1 == "add" ]]; **then** 72. **if** [ "$2" == "temp" ]; **then** #Добавление расширения в список расширений временных файлов 73. ext\_temp\_new="$ext\_temp ""$3" 74. change\_myconfig "$ext\_temp\_new" "$ext\_work" "$work\_folder" "$cur\_command" 75. **elif** [ "$2" == "work" ]; **then** #Добавление расширения в список расширений рабочих файлов 76. ext\_work\_new="$ext\_work ""$3" 77. change\_myconfig "$ext\_temp" "$ext\_work\_new" "$work\_folder" "$cur\_command" 78. **fi** 79. **elif** [[ $1 == "del" ]]; **then** 80. **if** [ "$2" == "temp" ]; **then** 81. **if** [ "$3" == "-a" ]; **then** #Удаление всех временных файлов в рабочей папке 82. **for** i **in** $ext\_temp; **do** 83. find "$work\_folder" -name "\*.$i" -delete 84. **done** 85. **else** #Удаление конкретного расширения из списка расширений временных файлов 86. r="\b""$3""\b" 87. ext\_temp\_new=$(echo "$ext\_temp" | sed -E s/"$r"//g) 88. change\_myconfig "$ext\_temp\_new" "$ext\_work" "$work\_folder" "$cur\_command" 89. **fi** 90. **elif** [ "$2" == "work" ]; **then** #Удаление конкретного расширения из списка расширений рабочих файлов 91. r="\b""$3""\b" 92. ext\_work\_new=$(echo "$ext\_work" | sed -E s/"$r"//g) 93. change\_myconfig "$ext\_temp" "$ext\_work\_new" "$work\_folder" "$cur\_command" 94. **elif** [ "$2" == "config" ]; **then** #Удаление файла, содержащего настройки скрипта 95. rm "$script\_folder"".myconfig" 96. **fi** 97. **elif** [[ $1 == "reload" ]]; **then** 98. **if** [ "$2" == "temp" ]; **then** #Задать заново список расширений временных файлов 99. ext\_temp\_new="" 100. **for** i **in** "$@"; **do** 101. **if** [ "$i" == "$1" ] || [ "$i" == "$2" ]; **then** 102. **continue** 103. **else** 104. ext\_temp\_new="$ext\_temp\_new ""$i" 105. **fi** 106. **done** 107. change\_myconfig "$ext\_temp\_new" "$ext\_work" "$work\_folder" "$cur\_command" 108. **elif** [ "$2" == "work" ]; **then** #Задать заново список расширений рабочих файлов 109. ext\_work\_new="" 110. **for** i **in** "$@"; **do** 111. **if** [ "$i" == "$1" ] || [ "$i" == "$2" ]; **then** 112. **continue** 113. **else** 114. ext\_work\_new="$ext\_work\_new ""$i" 115. **fi** 116. **done** 117. change\_myconfig "$ext\_temp" "$ext\_work\_new" "$work\_folder" "$cur\_command" 118. **elif** [ "$2" == "folder" ]; **then** #Изменение рабочей папки скрипта 119. change\_myconfig "$ext\_temp" "$ext\_work" "$3" "$cur\_command" 120. **elif** [ "$2" == "command" ]; **then** #Изменение записанной команды скрипта 121. new\_command="" 122. **for** i **in** "$@"; **do** 123. **if** [ "$i" == "$1" ] || [ "$i" == "$2" ]; **then** 124. **continue** 125. **else** 126. new\_command="$new\_command ""$i" 127. **fi** 128. **done** 129. change\_myconfig "$ext\_temp" "$ext\_work" "$work\_folder" "$new\_command" 130. **fi** 131. **elif** [[ $1 == "do" ]]; **then** #Исполнение записанной команды скрипта 132. eval "$cur\_command" 133. **else** 134. **while** :; **do** 135. echo -e "\n=======================================" 136. #Вывод списка расширений всех временных файлов 137. echo -e "\033[43m\033[30mСписок расширений временных файлов:\033[37m\033[0m\n" 138. sch=0 139. **for** i **in** $ext\_temp; **do** 140. sch=$((sch+1)) 141. echo $sch" расширение - \*.""$i" 142. **done** 143. #Вывод списка расширений всех рабочих файлов 144. echo -e "\n\033[43m\033[30mСписок расширений рабочих файлов:\033[37m\033[0m\n" 145. sch=0 146. **for** i **in** $ext\_work; **do** 147. sch=$((sch+1)) 148. echo $sch" расширение - \*.""$i" 149. **done** 150. #Вывод текущей записанной команды скрипта 151. echo -e "\n\033[43m\033[30mЗаписанная команда скрипта:\033[37m\033[0m \033[36m<<\033[37m$cur\_command\033[36m>>\033[37m\n" 152. #Вывод текущей рабочей папки скрипта 153. echo -e "\n\033[43m\033[30mТекущая рабочая папка скрипта:\033[37m\033[0m $work\_folder" 154. #Вывод интерфейса меню 155. echo -e "\n\033[33m1. Задать заново список временных файлов 156. 2. Добавить или удалить конкретное расширение из списка временных файлов 157. 3. Задать заново список рабочих файлов 158. 4. Добавить или удалить конкретное расширение из списка рабочих файлов 159. 5. Просмотреть или задать заново рабочую папку скрипта 160. 6. Удалить временные файлы 161. 7. Выполнить или изменить записанную команду 162. 8. Просмотреть число строк и слов в каждом рабочем файле 163. 9. Просмотреть объем каждого мусорного файла 164. 0. Выход из меню\033[37m\n" 165. read -rp "Введите желаемый номер меню: " n #Ввод желаемого номера пункта пользователем 166. **if** [[ $n == "1" ]]; **then** #Задать заново список расширений временных файлов 167. read -rp "Введите название нового расширения файлов: " name 168. ext\_temp="$name" 169. **while** :; **do** 170. read -rp "Введите название нового списка расширений или нажмите \"Enter\" для выхода: " name 171. **if** [ -n "$name" ]; **then** 172. ext\_temp="$ext\_temp"" ""$name" 173. **else** 174. **break** 175. **fi** 176. **done** 177. echo -e "\n\033[33m\_\_\_\_\_\_\_Список задан заново\_\_\_\_\_\_\_\033[37m\n" 178. **elif** [[ $n == "2" ]]; **then** 179. echo "Выберите необходимый пункт: 180. 1 -- Добавить конкретное расширение 181. 2 -- Удалить конкретное расширение (по значению)" 182. read -rp ": " number 183. **if** [[ $number == "1" ]]; **then** #Добавление конкретного расширения в список временных расширений 184. read -rp "Введите расширение, которое вы хотите добавить: " ext\_add 185. ext\_temp="$ext\_temp""$ext\_add " 186. echo -e "\n\033[33m\_\_\_\_\_\_\_Расширение добавлено\_\_\_\_\_\_\_\033[37m\n" 187. **elif** [[ $number == "2" ]]; **then** #Удаление конкретного расширения из списка временных расширений 188. read -rp "Введите расширение, которое вы хотите удалить: " ext\_rem 189. r="\b""$ext\_rem""\b" 190. ext\_temp=$(echo "$ext\_temp" | sed -E s/"$r"//g) 191. echo -e "\n\033[33m\_\_\_\_\_\_\_Расширение удалено\_\_\_\_\_\_\_\033[37m\n" 192. **fi** 193. **elif** [[ $n == "3" ]]; **then** #Задать заново список расширений временных файлов 194. read -rp "Введите название нового расширения файлов: " name 195. ext\_work="$name" 196. **while** :; **do** 197. read -rp "Введите название нового списка расширений или \"Enter\" для выхода: " name 198. **if** [ -n "$name" ]; **then** 199. ext\_work="$ext\_work"" ""$name" 200. **else** 201. **break** 202. **fi** 203. **done** 204. echo -e "\n\033[33m\_\_\_\_\_\_\_Список задан заново\_\_\_\_\_\_\_\033[37m\n" 205. **elif** [[ $n == "4" ]]; **then** 206. echo "Выберите необходимый пункт: 207. 1 -- Добавить конкретное расширение 208. 2 -- Удалить конкретное расширение (по значению)" 209. read -rp ": " number 210. **if** [[ $number == "1" ]]; **then** #Добавление расширения в список расширений рабочих файлов 211. read -rp "Введите расширение, которое вы хотите добавить: " ext\_add 212. ext\_work="$ext\_work""$ext\_add " 213. echo -e "\n\033[33m\_\_\_\_\_\_\_Расширение добавлено\_\_\_\_\_\_\_\033[37m\n" 214. **elif** [[ $number == "2" ]]; **then** #Удаление конкретного расширения из списка расширений рабочих файлов 215. read -rp "Введите расширение, которое вы хотите удалить: " ext\_rem 216. r="\b""$ext\_rem""\b" 217. ext\_work=$(echo "$ext\_work" | sed -E s/"$r"//g) 218. echo -e "\n\033[33m\_\_\_\_\_\_\_Расширение удалено\_\_\_\_\_\_\_\033[37m\n" 219. **fi** 220. **elif** [[ $n == "5" ]]; **then** 221. echo "Выберите необходимый пункт: 222. 1 -- Просмотреть рабочую папку 223. 2 -- Задать заново рабочую папку" 224. read -rp ": " number 225. **if** [[ $number == "1" ]]; **then** #Просмотр содержимого рабочей папки 226. echo -e "\n\033[33mСодержимое рабочей папки:\033[37m \n""$(ls "$work\_folder")" 227. **elif** [[ $number == "2" ]]; **then** #Задать заново рабочую папку 228. read -rp "Введите путь к новой рабочей папке: " work\_folder 229. work\_folder=${work\_folder//'~'/$HOME} 230. echo -e "\n\033[33m\_\_\_\_\_\_\_Рабочая папка задана\_\_\_\_\_\_\_\033[37m\n" 231. **fi** 232. **elif** [[ $n == "6" ]]; **then** #Удаление всех временных файлов в рабочей папке 233. **for** i **in** $ext\_temp; **do** 234. find "$work\_folder" -name "\*.$i" -delete 235. **done** 236. echo -e "\n\033[33m\_\_\_\_\_\_\_Временные файлы удалены\_\_\_\_\_\_\_\033[37m\n" 237. **elif** [[ $n == "7" ]]; **then** 238. echo "Выберите необходимый пункт: 239. 1 -- Выполнить записанную команду 240. 2 -- Изменить записанную команду" 241. read -rp ": " number 242. **if** [[ $number == "1" ]]; **then** #Выполнение записанной команды скрипта 243. cur=$(pwd) 244. cd "$work\_folder" || exit 245. eval "$cur\_command" 246. cd "$cur/" || exit 247. echo -e "\n\033[33m\_\_\_\_\_\_\_Команда выполнена\_\_\_\_\_\_\_\033[37m\n" 248. **elif** [[ $number == "2" ]]; **then** #Изменение записанной команды скрипта 249. read -rp "Введите новую команду: " cur\_command 250. echo -e "\n\033[33m\_\_\_\_\_\_\_Команда изменена\_\_\_\_\_\_\_\033[37m\n" 251. **fi** 252. **elif** [[ $n == "8" ]]; **then** #Вывод числа строк и слов в каждом рабочем файле в рабочей папке 253. **for** i **in** $ext\_work; **do** 254. str=$(ls "$work\_folder" | grep \."$i") 255. **if** [[ $str =~ .+ ]]; **then** 256. **for** j **in** "$work\_folder"\*."$i"; **do** 257. echo -e "\033[33mВ файле\033[37m $j\033[33m:\033[37m $(wc -l < "$j") \033[33mстрок и\033[37m $(wc -w < "$j") \033[33mслов\033[37m" 258. **done** 259. **fi** 260. **done** 261. **elif** [[ $n == "9" ]]; **then** #Вывод размера всех мусорных файлов в рабочей папке 262. **for** i **in** $ext\_temp; **do** 263. str=$(ls "$work\_folder" | grep \."$i") 264. **if** [[ $str =~ .+ ]]; **then** 265. **for** j **in** "$work\_folder"\*."$i"; **do** 266. echo -e "\033[33mФайл\033[37m $j \033[33mимеет размер:\033[37m $(wc -c < "$j") \033[33mбайтов\033[37m" 267. **done** 268. **fi** 269. **done** 270. **elif** [[ $n == "0" ]]; **then** #Выход из скрипта 271. echo -e "\n\033[31mВыберите желаемый пункт: 272. 1 -- Выйти из меню 273. 2 -- Удалить файл с настройками скрипта и выйти из меню\033[37m" 274. read -rp ": " number 275. **if** [[ $number == "1" ]]; **then** 276. exit 277. **elif** [[ $number == "2" ]]; **then** 278. rm "$script\_folder"".myconfig" 279. exit 280. **fi** 281. **fi** 282. **done** 283. **fi** |