|  |  |
| --- | --- |
|  | **Отчет по лабораторной работе № 21**  Студент группы М8О-103Б-23 Махмутов Дэниз Ирикович, № по списку 17  Контакты www, e-mail, icq, skype lustmodesta@gmail.com  Работа выполнена: « 6» апреля 2024 г.  Преподаватель: доцент каф. 806 Никулин С.П.  Входной контроль знаний с оценкой  Отчет сдан « » 2024 г., итоговая оценка  Подпись преподавателя |

1. **Тема:** Программирование на интерпретируемых командных языках
2. **Цель работы:** составить программу выполнения заданных действий над файлами на одном из интерпретируемых командных языков ОС UNIX

3**. Задание (***вариант №* 12 **):** генерация заданного числа копий указанного файла, Имена копий генерировать добавлением к имени исходного файла следующей по порядку буквы или цифры, начиная с заданной буквы или цифры (29-17)

**Оборудование** (*лабораторное*):

Оборудование (лабораторное):

ЭВМ , процессор , имя узла сети c ОП Мб,

НМД Мб. Терминал адрес . Принтер

Другие устройства

Оборудование ПЭВМ студента, если использовалось:

Процессор AMD Ryzen 5 5600x, 6 Cores с ОП Ubuntu 22.04.3 LTS Мб,

НМД 486,400 Мб. Монитор

Другие устройства

1. **Программное обеспечение** (*лабораторное*):

Операционная система семейства , наименование версия

Интерпретатор команд bash версия

Система программирования gcc версия

Редактор текстов версия

Утилиты операционной системы

Прикладные системы и программы

Местонахождения и имена файлов программ и данных

*Программное обеспечение ЭВМ студента, если использовалось:*

*Операционная система семейства Linux, наименование Ubuntu версия 22.04.3 LTS*

*интерпретатор команд GNU bash версия 5.1.16(1)-release (x86/64-pc-linux-gnu)*

*Система программирования C версия 2.35*

*Редактор текстов GNU emacs версия 27.1*

1. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальное описание с пред- и постусловиями)

Задание сводится к тому, чтобы использовать команду cp для копирования файла определенное количество раз. Однако нужно также учитывать, куда добавить номер: если файл не имеет расширения, номер будет добавлен в конец имени файла, а если у файла есть расширение, номер будет добавлен перед расширением. Для этого нужно разделить название файла на имя и расширение. Также хорошей идеей будет сделать ввод данных параметрическим, чтобы программа запрашивала данные у пользователя после вызова скрипта. Если пользователь не введет необходимые параметры, программа должна запросить их. Для обработки ситуации, когда не все параметры введены, можно предусмотреть стандартные значения переменных.

1. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты, либо соображения по тестированию].

Программа проверяет, были ли введены параметры. Если нет, программа запрашивает их у пользователя. Также, если был введен один параметр, программа проверяет, был ли введен второй параметр. Если нет, то она записывает предопределенное значение в соответствующую переменную (количество копий и начальная букве, так как скрипт работает с файлами). Потом основная часть алгоритма, где мы используем cut для проверки содержимого файла и наличия у него расширения. Передаем в cut название файла, чтобы разделить его по точке и получить две переменные. После разделения возможны два варианта: 1) если переменные разные, это означает, что в файле есть расширение, и мы создаем копии, добавляя номер копии перед расширением. 2) если переменные одинаковые (это может произойти, если cut вернет последнее поле, если в слове n полей меньше запрашиваемого количества), это означает, что в файле нет расширения, и мы просто добавляем букву в конец имени файла.

*Пункты 1-7 отчёта составляются* ***строго до*** *начала лабораторной работы.*

Допущен к выполнению работы. Подпись преподавателя

1. **Распечатка протокола** (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем)

|  |
| --- |
| unix@DESKTOP-MPQDBS2:~/labs/21$ cat main.sh  #!/bin/bash  if [[ -z $1 ]]; then  echo -n "Введите название файла : "  read name\_of\_file  echo -n "Введите кол-во копий : "  read number\_of\_copy  echo -n "Введите цифру, с которой будут начинаться номера копий : "  read first\_number\_of\_copy  else  name\_of\_file=$1  number\_of\_copy=$2  first\_number\_of\_copy=$3  if [[ -z $2 ]]; then  number\_of\_copy=1  fi  if [[ -z $3 ]]; then  first\_number\_of\_copy=1  fi  fi  let "first\_number\_of\_copy = first\_number\_of\_copy - 1"  let "number\_of\_copy = number\_of\_copy + first\_number\_of\_copy"  end\_of\_file=`echo $name\_of\_file | cut -f 2 -d '.'`  start\_of\_file=`echo $name\_of\_file | cut -f 1 -d '.'`  if [[ "$start\_of\_file" == "$end\_of\_file" ]]; then  while [[ $number\_of\_copy > $first\_number\_of\_copy ]]  do  cp $name\_of\_file $start\_of\_file"\_"$number\_of\_copy  let "number\_of\_copy = number\_of\_copy - 1"  done  else  while [[ $number\_of\_copy > $first\_number\_of\_copy ]]  do  cp $name\_of\_file $start\_of\_file"\_"$number\_of\_copy"."$end\_of\_file  let "number\_of\_copy = number\_of\_copy - 1"  done  fi  exit 0  unix@DESKTOP-MPQDBS2:~/labs/21$ cat JustFile  It's just a file with a very simple text  It's gonna be used to test utlite  Let's go!  unix@DESKTOP-MPQDBS2:~/labs/21$ cat TextScript.sh  #!/bin/bash  echo "Hello world!"  unix@DESKTOP-MPQDBS2:~/labs/21$ cat Dir/FILE2  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh  Введите название файла : JustFile  Введите кол-во копий : 2  Введите цифру, с которой будут начинаться номера копий : 2  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  1\_3.txt 1\_4.txt 1.txt Dir JustFile JustFile\_2 JustFile\_3 JustFile\_4 main.sh prot21.txt TextScript.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ rm \*\_\*  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  1.txt Dir JustFile main.sh prot21.txt TextScript.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh  Введите название файла : ^Z  [3]+ Stopped ./main.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ rm 1.txt  unix@DESKTOP-MPQDBS2:~/labs/21$  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh  Введите название файла : JustFile  Введите кол-во копий : 2  Введите цифру, с которой будут начинаться номера копий : 2  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  Dir JustFile JustFile\_2 JustFile\_3 main.sh prot21.txt TextScript.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh TextScript.sh 3 4  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  Dir JustFile JustFile\_2 JustFile\_3 main.sh prot21.txt TextScript\_4.sh TextScript\_5.sh TextScript\_6.sh TextScript.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh Dir/FILE2 1 1  unix@DESKTOP-MPQDBS2:~/labs/21$ ls Dir/  FILE2 FILE2\_1  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh JustFile\_2 2  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  Dir JustFile JustFile\_2 JustFile\_2\_1 JustFile\_2\_2 JustFile\_3 main.sh prot21.txt TextScript\_4.sh TextScript\_5.sh TextScript\_6.sh TextScript.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ ./main.sh TextScript\_5.sh  unix@DESKTOP-MPQDBS2:~/labs/21$ ls  Dir JustFile JustFile\_2 JustFile\_2\_1 JustFile\_2\_2 JustFile\_3 main.sh prot21.txt TextScript\_4.sh TextScript\_5\_1.sh TextScript\_5.sh TextScript\_6.sh TextScript.sh |

1. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб.  или  дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  |  |  |  |  |  |  |

1. Замечание автора по существу работы
2. Выводы смогли составить программу выполнения заданных действий над файлами на одном из интерпретируемых языков ОС UNIX, разобрали структуру интерпретируемых языков, способы программирования скриптов.

Недочеты, допущенные при выполнении задания, могут быть устранены следующим образом

Подпись студента