Отчет по лабораторной работе №12.

Программирование в командном процессоре ОС UNIX. Расширенное программирование

Данила Андреевич Стариков

Содержание

# 1 Цель работы

Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# 2 Выполнение лабораторной работы

## 2.1 Задание 1

Задание: Реализовать команду man с помощью командного файла. Изучите содержимое каталога /usr/share/man/man1. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой less сразу же просмотрев содержимое справки. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге man1.

Создан командный файл manlike.sh (Листинг 1). Примеры выполнения скрипта представлены на Рисунках 1 и 2 .

Листинг 1: Текст командного файла Задания №1.

#!/bin/bash  
COMMAND=$1  
FILE=`find /usr/share/man/man1 -type f -name $COMMAND.1.gz -print -quit`  
if [ -n "$FILE" ]  
then   
 less -R /usr/share/man/man1/$COMMAND.1.gz 2>/dev/null  
else   
 echo command not found  
fi  
OPTIONS+="-e $pval $ival"   
if [ ! $Cflag ]  
then OPTIONS+=" -i"  
fi  
if [ $nflag ]  
then OPTIONS+=" -n"  
fi  
grep ${OPTIONS} > $oval

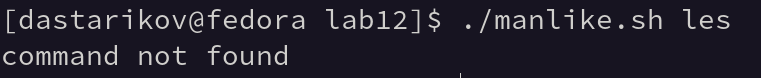


Рис. 1: Пример №1 работы программы manlike.sh.

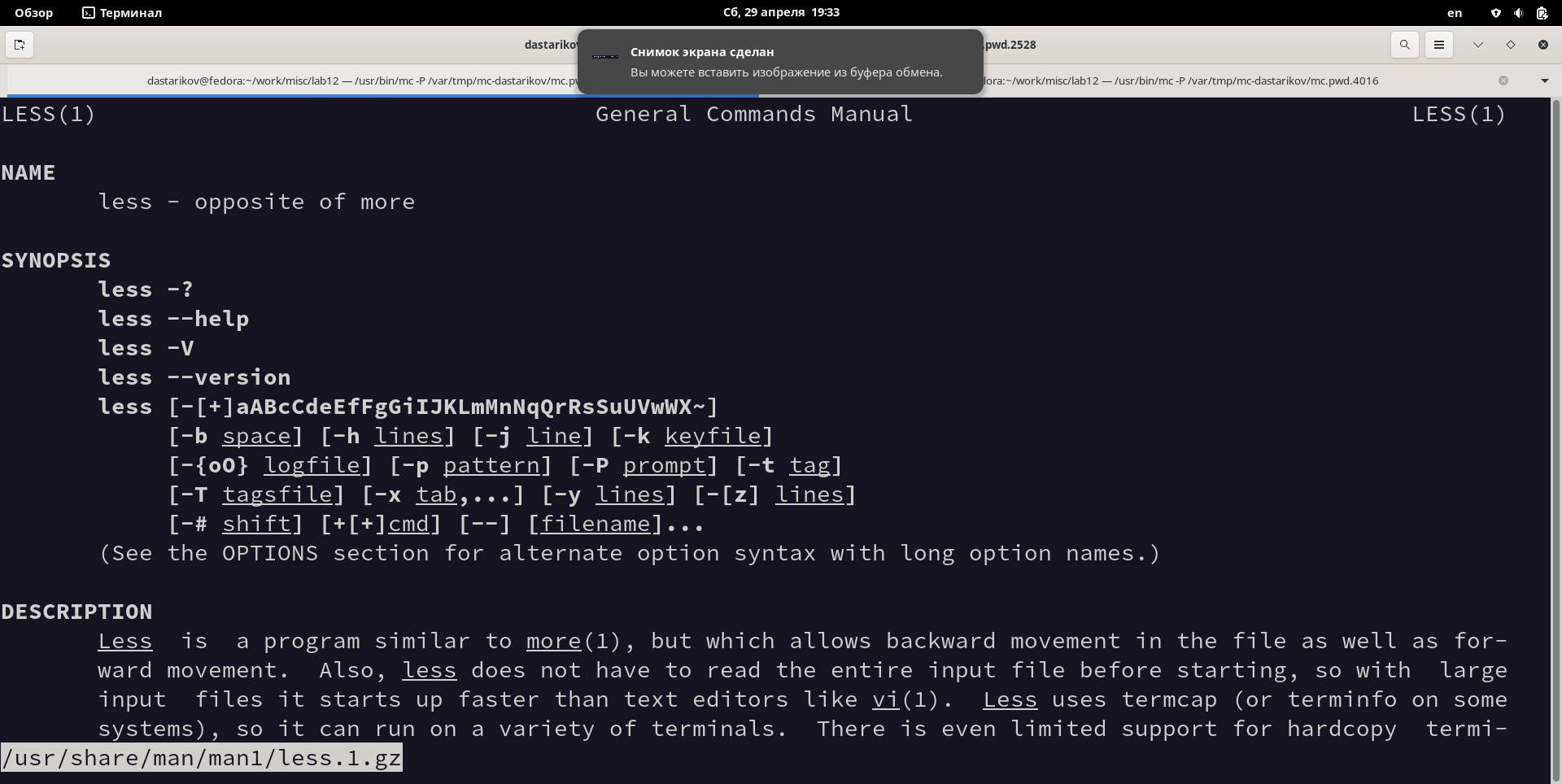


Рис. 2: Пример №2 работы программы manlike.sh.

## 2.2 Задание 2

Задание: Используя встроенную переменную $RANDOM, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что $RANDOM выдаёт псевдослучайные числа в диапазоне от 0 до 32767.

Создан командный файл randchar.sh (Листинг 2). Примеры выполнения скрипта представлены на Рисунке 3.

Листинг 2: Текст командного файла Задания №2.

#!/bin/bash  
let NUMBER=$RANDOM  
while [ ! $NUMBER -eq 0 ]  
do  
 let VALUE=33+$NUMBER%94  
 echo -ne "\x$(printf %02x $VALUE)"  
 let NUMBER=$NUMBER/2  
done  
echo

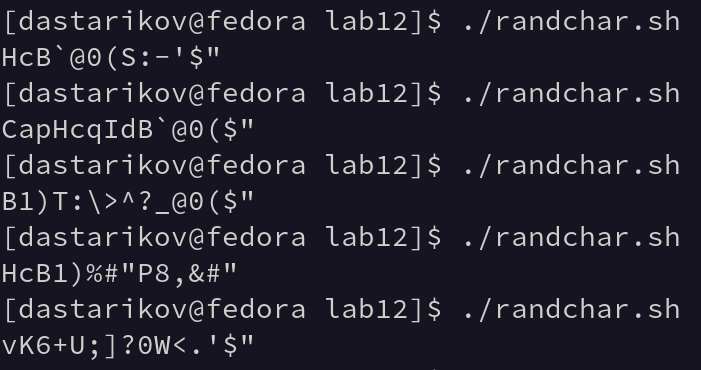


Рис. 3: Пример работы программы randchar.sh.

# 3 Выводы

В рамках лабораторной работы изучены основы программирования в оболочке ОС UNIX. Получены навыки написания более сложных командных файлов с использованием логических управляющих конструкций и циклов.