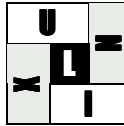


**Институт транспорта и связи**

**Шамшин Ю.В.**

**Методические указания к выполнению домашней работы N2  
по курсу  
«Операционные системы»**



**Программирование в shell  
и подготовка инсталляционных пакетов  
в ОС UNIX/Linux**

**Рига 2003**

## **Цель работы**

- научиться осмысленно читать имеющиеся в системе сценарии;
- приобрести навыки по составлению и отладке собственных сценариев на языке командного интерпретатора UNIX/Linux;
- получить основы технологий создания сопроводительной документации;
- закрепить навыки по работе с программами архивирования и сжатия файлов в UNIX/Linux (tar, gzip, gunzip);
- познакомиться с основами компиляции, компоновки и инсталляции программ (gcc, make, install, rpm);
- отработка навыков работы в проектной группе;
- проверка умения решать нестандартные задачи.

## **Порядок выполнения домашнего задания**

### **1. Получить вариант домашнего задания.**

Образовать проектную группу из 2-3 человек. Распределить роли. Получить у преподавателя индивидуальное домашнее задание. Пример варианта задания приведен в приложении 1.

### **2. Написать и отладить минимальный сценарий.**

Разработать алгоритм, реализующий выполнение поставленной задачи при минимуме функций и «украшений». Напишите минимальный сценарий и проверьте придуманный вами алгоритм в действии. Придумайте компактное смысловое имя для созданного сценария.

### **3. Расширить функциональность сценария.**

Пишется окончательный вид сценария и проводится его доработка. Окончательный сценарий должен иметь следующие функции и обеспечивать следующие характеристики:

- поддерживать различные коды завершения (exit 0, exit 1, ...);
- выводить краткую справку при неверном формате вызова сценария (например, “Use format: scriptname [-m] <filename1>[ <filename2>]...”);
- иметь не менее двух простых ключей (например: -h – help, -a – author, -v – version);
- иметь как минимум один сложный ключ, модифицирующий алгоритм исполнения сценария (например, как ключ -l в команде ls -l);
- поддерживать при выводе краткой справки и/или в ходе диалога с пользователем эффекты управления цветом и/или курсором;
- уметь в ходе диалога с пользователем выводить справку (помощь) при нажатии клавиши <F1> и/или прерываться при нажатии клавиши <F10>;
- обрабатывать сигналы (trap).

Пример сценария приведен в приложении 2.

### **4. Оформить готовый сценарий надлежащим образом.**

Каждый сценарий должен иметь следующую структуру (см. приложение 2):

- magic-последовательность (например, #!/bin/sh);
- шапка (содержит: краткое и длинное имя сценария, его назначение, авторство, дату, версию);
- описание работы (ключи, коды возврата);
- сам сценарий с необходимыми комментариями.

## 5. Создать страницу помощи для вашего сценария в формате man.

Пример man-страницы приведен в приложении 3. Man-страница должна содержать следующие разделы:

- NAME – имя и функция;
- SYNOPSIS – синтаксис;
- DESCRIPTION – описание функции;
- FILE – используемые файлы;
- OPTIONS – описание модификаторов команды;
- DIAGNOSTIC – ответы на ошибки;
- BUGS – замеченные некорректности;
- AUTHOR – авторы man-страницы;
- SEE ALSO – смежные команды.

## 6. Подготовить инсталляционный файл Makefile.

Требования к инсталляции:

- В файле Makefile должны присутствовать секции install, delete, clear.
- Инсталляция должна по умолчанию проводиться в каталог ~/bin/SCRIPT-NAME текущего пользователя.
- Процесс инсталляции не должен требовать прав супер-пользователя для установки и удаления вашего пакета («тарболла»).
- Сценарий должен уметь исполняться от имени обычного пользователя.

Пример man-страницы приведен в приложении 4.

## 7. Оформить отчет по домашнему заданию.

Отчет оформляется на русском, латышском или английском языках в формате TXT, RTF, PDF, PS, HTML.

Отчет должен содержать:

- Титульный лист, см. приложение 5.
- Задание на лабораторную работу.
- Описание созданного сценария с возможными копиями экрана.
- Текст сценария с необходимыми комментариями.
- Состав инсталляционного пакета (список файлов: имена, назначение, размер).
- Краткие указания по установке (инсталляции) и удалению программы.
- Выводы по работе, возможное улучшение сценария, области его применения.
- Используемая литература и источники.

## 8. Подготовить инсталляционный пакет сценария.

Из созданных ранее компонентов необходимо приготовить инсталляционный пакет SCRIPT-NAME.TAR.GZ в виде сжатого архива. Инсталляционный пакет должен содержать:

- ваш готовый сценарий,
- файл помощи в формате man-страницы,
- Makefile (конфигурационный файл для команды make),
- файл README (краткие пояснения по программе, формат ввода-вывода, авторство),
- файл INSTALL (описание процедуры инсталляции и деинсталляции),
- другие необходимые файлы (возможные библиотеки, файлы данных, тестовые файлы и прочее).

Примечание. Файл SCRIPT-NAME.TAR.GZ должен распаковываться в отдельную одноименную с архивом директорию в текущем каталоге, т.е. в папку ./SCRIPT-NAME.

## 9. Подготовить электронный архив домашнего задания.

Домашнее задание представляется для проверки в электронном виде как один архивный файл ZIP на дискете или по электронной почте на указанный преподавателем адрес.

Архив должен иметь имя в формате GRUPPA-FAMILIJA (например, 4901BV-PUPKIN.ZIP) и распаковываться в одноименный каталог в текущей директории.

Архив должен содержать:

- REPORT.PDF – Текст отчета в формате TXT, RTF, PDF, PS, HTML.
- SCRIPT-NAME.TAR.GZ – Инсталляционный пакет в виде сжатого архива («тарболла»).
- MD5SUM для предыдущих двух файлов.

## 10. Защита работы.

Домашнее задание представляется для защиты в напечатанном и электронном виде с демонстрацией работы инсталляции, работы и деинсталляции созданной script-программы.

### *Рекомендуемая литература*

1. Шамшин Ю.В. ОС UNIX: Лабораторный практикум. Часть I. –Рига: TSI, 2003. –144 с.
2. Григорий Строкин. BASH конспект. Копия на <http://www.junik.lv/~georg/unixlearn/bash-conspect.htm>
3. UNIXhelp home page. <http://www.ucs.ed.ac.uk/~unixhelp/index.html>
4. Тейнсли Д. Linux и UNIX: программирование в shell. Руководство разработчика: Пер. с англ. –К.: BHV, 2001. – 464 с.
5. Mendel Cooper. Advanced Bash Scripting Guide (Искусство программирования на языке сценариев командной оболочки). <http://personal.riverusers.com/~thegrendel/>

Пример варианта индивидуального задания

Написать простой калькулятор на shell, умеющий выполнять арифметические действия, работать с процентами и целыми степенями. Используйте case, echo, read, expr, printf.

Пример оформления сценария

Конвертер файлов из формата DOS (новая строка в виде двух символов) в формат UNIX (переход на новую строку осуществляется одним символом).

```
#!/bin/bash
#####
# Shellscrip:      dtox - DOS to UNIX text file conversion
# Author       :   Heiner Steven <heiner.steven@odn.de>
# Category    :   File Conversion
# SCCS-Id.    :   @(#) dtox 1.1 02/02/26
#####
# Description
#       Replaces "CR LF" sequences with "LF" end-of-line.
#       Does not handle DOS end-of-file character CTRL-Z (ASCII 26).
#####

PN=`basename "$0"`          # Program name
VER='1.1'

Usage () {
    echo >&2 "$PN - DOS to UNIX text file conversion, $VER
usage: $PN file [file ...]"
    exit 1
}

Msg () {
    for MsgLine
    do echo "$PN: $MsgLine" >&2
    done
}

Fatal () { Msg "$@"; exit 1; }

set -- `getopt h "$@"` || Usage
[ $# -lt 1 ] && Usage          # "getopt" detected an error

while [ $# -gt 0 ]
do
    case "$1" in
        --)      shift; break;;          # your flags here
        -h)      Usage;;
        -*)      Usage;;
        *)      break;;                 # First file name
    esac
    shift
done

[ $# -lt 1 ] && Usage

set -e                        # Terminate at first error
for file
do
    echo >&2 "$file"
    sed 's/
$//' < "$file" > tmp.$$ &&
    mv tmp.$$ "$file"
done
```

**Пример структуры и содержимого man-страницы**

Файл называется hob2d.1

```
.TH MYSCRIPT 1 "HOB2D Useless Utilities" "PupkinNET" \" -*- nroff -*-
.SH NAME
hob2d \- convert Hex, Oct or Bin number(s) to Dig number(s)
.SH SYNOPSIS
.B hob2d [-h|-o|-b] [number]
.SH DESCRIPTION
This manual page
documents the swell version of
.B hob2d.
.B hob2d
accepts one command\line argument, a 3 digit number in the form of a
hex, oct or bin. The result is printed to the console and is the
digital format of the one entered on the command line.
.SH OPTIONS
.B -e
stands for hElp.
SH BUGS
Sometimes prefixing a number with zeroes causes errors?!?!
(ie. hob2d -h 01a -> 126, but hob2d -h 1a -> 026)
.SH COPYRIGHT
Copyright 2003, 2004 Vasja Pupkin.

hob2d is free software; you can distribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software
Foundation.

.SH AUTHOR
Vasjanja

.SH SEE ALSO
bc(1)
```

**Приложение 4.****Пример структуры и содержимого Makefile**

```
#
# Makefile for calc - a rudimentary emulation of a HP28S
#
RM      = rm -f
CC      = gcc
LIBS    = -lm
# Added -ggdb for debugging
#CFLAGS = -O6 -Wall -ggdb
CFLAGS  = -O6 -Wall
DEFINES =

PROG    = calc
SRC     = $(PROG).c
HEADER  = $(PROG).h

MANSEC  = 1
MAN     = $(PROG).$(MANSEC)
DOC     = README.1st README README.Linux COPYING calc-1.1.lsm
INSTDIR = /usr/local
BINDIR  = $(INSTDIR)/bin
MANDIR  = $(INSTDIR)/man/man$(MANSEC)

calc:   calc.h calc.c
        $(CC) $(CFLAGS) $(SRC) -o $(PROG) $(LIBS) $(DEFINES)

clean:
        $(RM) $(PROG) *.o *~ *.bak errors calc core

install:
        install -g users -o bin -m 755 -s $(PROG) $(BINDIR)
install.man:
        install -g info -o bin -m 644 $(MAN) $(MANDIR)

tar:
        tar cvfz $(PROG).tgz $(SRC) $(HEADER) $(MAN) Makefile $(DOC)
```

**Институт транспорта и связи  
«Операционные системы»**

**Отчет  
по домашней работе N 2  
Программирование в shell  
и подготовка инсталляционных пакетов  
в ОС UNIX/Linux**

**Выполнили:  
студенты гр. 4321 ВД  
Василий Пупкин  
Масяня Иванова  
Янчик Озолс**

**Проверил:  
доцент каф. компьютерных сетей, Dr.Sc. Eng.  
Юрий Шамшин**

**Рига 2003**