

Санкт-Петербургский Национально Исследовательский Университет
информационных технологий, механики и оптики
Кафедра систем управления и информатики

Системное программное обеспечение

Отчет по лабораторной работе №3

Написание скриптов на bash

Вариант №1

Работу

выполнили:

Зенкин А.М.

Карпов К.В.

Пачковский К.А.

Группа: Р3335

Преподаватель:

Мусаев А.А.

Санкт-Петербург
2017

Содержание

1. Цель работы	2
2. Ход работы	2
2.1. Задание 1 «Написание скриптов. Операторы ветвления»	2
2.1.1. Знакомство с командной оболочкой bash	2
2.1.2. Bash-скрипт, который выводит Ф.И.О. и вариант группы тремя способами:	2
2.1.3. Анализ каждого метода из пункта 2.1.2:	3
2.1.4. Использование операторов ветвления:	3
2.2. Задание 2 «Написание скриптов на bash. Циклы»	4
2.2.1. Использование циклов:	4
2.2.2. Факториал n для заданного n , принадлежащего \mathbb{R} :	4
3. Вывод	5

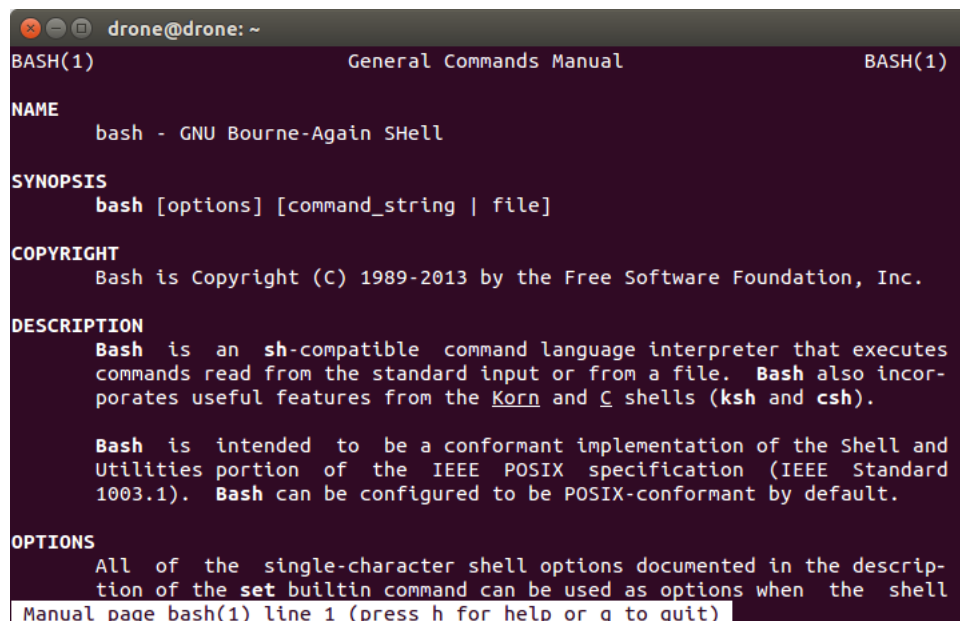
1. Цель работы

Написание скриптов на bash.

2. Ход работы

2.1. Задание 1 «Написание скриптов. Операторы ветвления»

2.1.1. Знакомство с командной оболочкой bash



```
drone@drone: ~
BASH(1)                                General Commands Manual                                BASH(1)

NAME
    bash - GNU Bourne-Again SHell

SYNOPSIS
    bash [options] [command_string | file]

COPYRIGHT
    Bash is Copyright (C) 1989-2013 by the Free Software Foundation, Inc.

DESCRIPTION
    Bash is an sh-compatible command language interpreter that executes
    commands read from the standard input or from a file.  Bash also incor-
    porates useful features from the Korn and C shells (ksh and csh).

    Bash is intended to be a conformant implementation of the Shell and
    Utilities portion of the IEEE POSIX specification (IEEE Standard
    1003.1).  Bash can be configured to be POSIX-conformant by default.

OPTIONS
    All of the single-character shell options documented in the descrip-
    tion of the set builtin command can be used as options when the shell
    is invoked.

Manual page bash(1) line 1 (press h for help or q to quit)
```

Рисунок 2.1. - Отображаем справку по bash

2.1.2. Bash-скрипт, который выводит Ф.И.О. и вариант группы тремя спосо- бами:

Программа 1: Ф.И.О

```
1 #!/bin/bash
2
3 # First method
4 echo "Zenkin A.M., Karpov K.V., Pachkovsky K.A. Variant:1"
5
6 #Second method
7 nameVar="Zenkin A.M., Karpov K.V., Pachkovsky K.A. Variant:1"
8 echo $nameVar
9
10 #Fird method
11 printData() {
12     local data="Zenkin A.M., Karpov K.V., Pachkovsky K.A. Variant:1"
13     echo $data
14 }
15
16 printData
```

2.1.3. Анализ каждого метода из пункта 2.1.2:

1. Преимущество: код меньший по объёму. Недостатки: выводится текст с которым никак нельзя взаимодействовать.

2. Достоинства в том, что можно производить операции с этой строковой переменной. Недостаток, что занимает память.

3. Достоинство заключается в том, что можно использовать любое название переменной, т.к. она существует только внутри функции. Недостаток в том, что код может стать непонятным.

2.1.4. Использование операторов ветвления:

Программа 2: проверка на число

```
1 #!/bin/bash
2
3 myVar=1
4
5 echo "Enter var"
6 read data
7 if !(echo "$data" | grep -E -q "^[0-9]+$")
8 then
9     echo "Not Number, please try again"
10    read data
11 fi
12
13 if [ $data -eq $myVar ]
14 then
15     echo "Variant is correct"
16 fi
```

2.2. Задание 2 «Написание скриптов на bash. Циклы»

2.2.1. Использование циклов:

Программа 3: группы

```
1 #!/bin/bash
2
3 while [ 1 -eq 1 ]
4 do
5 echo "Enter your name: "
6 read name
7 if [ -z "$name" ]
8 then
9 break
10 fi
11 echo "The name you entered is '$name'. Hi $name!"
12 echo "Do you want to continue your work? Please, enter 'yes' or 'no'"
13 read answer
14 if [ "$answer" = "yes" ]
15 then
16 echo "Enter your age: "
17 read age
18 if !(echo "$age" | grep -E -q "^[0-9]+$")
19 then
20 break
21 fi
22 if [ "$age" -eq 0 ]
23 then break
24 elif [ $age -ge 1 -a $age -le 16 ]; then
25 echo "$name, your group is child"
26 elif [ $age -ge 17 -a $age -le 25 ]; then
27 echo "$name, your group is youth"
28 else echo "$name, your group is adult"
29 fi
30 else
31 break
32 fi
33 done
34 echo "Bye"
```

2.2.2. Факториал n для заданного n, принадлежащего R:

Программа 4: факториал

```
1 #!/bin/bash
2
3 echo "Enter number: "
4 read n
5 mult=1
6 for (( i=1; i<=n; i++ ))
7 do
8 let "mult=i*mult"
9 done
10 echo $mult
```

3. Вывод

В данной лабораторной работе были получены навыки в написании скриптов на bash - основ программирования, циклов и операторов ветвления.