

Университет ИТМО

Кафедра ВТ

Системное программное обеспечение
Лабораторная работа №6

Выполнил студент 2 курса
Группы Р3211 Романов Олег
Преподаватель: Дергачев А.М.

Санкт-Петербург
2016 год

Задание #1 (файл /usr/local/regexp/db)

1. выведите все номера телефонов

```
nawk -F: '{print $2}' db
```

2. выведите номер телефона, принадлежащий сотруднику Dan

```
nawk -F: '/Dan/{print $2}' db
```

3. выведите имя, фамилию и номер телефона сотрудницы Susan

```
nawk -F: '/Susan/{print $1,$2}' db
```

4. выведите все фамилии, начинающиеся с буквы D

```
nawk -F: '| ' '$2~/^D/{print $2}' db
```

5. выведите все имена, начинающиеся с буквы С или Е

```
nawk '/^[C|E]/{print $1}' db
```

6. выведите все имена, состоящие только из четырех букв

```
nawk '/^.... /{print $1}' db
```

7. выведите имена сотрудников, префикс номера телефона которых 916

```
nawk '/\ (916\)/{print $1}' db
```

8. выведите денежные вклады сотрудника Mike, предваряя каждую сумму знаком \$

```
nawk -F: '/Mike/{print $3"$"$4"$"$5}' db
```

9. выведите инициалы всех сотрудников

```
nawk '{ $1=substr($1,1,1); $2=substr($2,1,1); print $1"."$2"." }' db
```

10. создайте командный файл awk, который:

- печатает полные имена и номера телефонов всех сотрудников по фамилии Savage
- печатает денежные вклады сотрудника по имени Chet
- печатает сотрудников, денежные вклады которых в первом месяце составили 250\$
- подсчитывает сумму вкладов за каждый месяц в отдельности и вывести это в виде оформленной таблицы
- подсчитывает средний вклад за каждый месяц и выводит результаты округлённо до второго знака после запятой
- в конце вывести текущее время и результат выполнения команды ls

```
{if ($2 == "Savage") print $1, $2, $3, $4
if ($1~/^Chet/) print "CHet's money:" $5, $6, $7
if ($5 == "250") print $1, $2, $5
first += $5; second += $6; third += $7 }
END {printf "%20s\n", "Sum"; printf "%10s %10s %10s\n", "FIRST","SECOND","THIRD"
printf "%10s %10s %10s\n", first, second, third
printf "\n%20s\n", "Middle"; printf "%10s %10s %10s\n", "FIRST","SECOND","THIRD"
printf "%10.2f %10.2f %10.2f\n", first/NR, second/NR, third/NR;
system("date");system("ls");}
nawk -F: '| ' -f ~/prog db
```

Задание #2:

1. nawk '/west/' datafile

В результате работы программы будут выведены все строки, содержащие подстроку west

2. nawk '/^north/' datafile

В результате работы программы будут выведены все строки, начинающиеся с подстроки north

3. nawk '/^(no|so)/' datafile

В результате работы программы будут выведены все строки, начинающиеся с подстроки no или подстроки so

4. nawk '{print \$3, \$2}' datafile

В результате работы программы будут выведены через пробел 3 и 2 поля каждой строки

(Запятая указывает на необходимость использовать при выводе в качестве разделителя полей содержимое переменной OFS, значением по умолчанию которой является пробел)

```
5. awk '{print $3 $2}' datafile
```

В результате работы программы будут выведены подряд 3 и 2 поля каждой строки

```
6. awk '{print $0}' datafile
```

В результате работы программы будут выведены все строки указанного файла

```
7. awk '{print "Number of fields: "NF}' datafile
```

В результате работы программы для каждой строки исходного файла будет выведена строка Number of fields: , а затем кол-во полей в исходной строке

```
8. awk '/northeast/{print $3, $2}' datafile
```

В результате работы программы будут выведены 3 и 2 поля строк, содержащих подстроку northeast

```
9. awk '/E/' datafile
```

В результате работы программы будут выведены все строки, содержащие символ E

```
10. awk '/^[ns]/{print $1}' datafile
```

В результате работы программы будут выведены первые поля для строк, начинающихся с символа n или символа s

```
11. awk '$5 ~ /\.[7-9]+/' datafile
```

В результате работы программы будут выведены все строки, пятое поле которых содержит подстроку, начинающуюся с точки, за которой следует один или более символов 7, 8 или 9

```
12. awk '$2 !~ /E/{print $1, $2}' datafile
```

В результате работы программы будут выведены первое и второе поля через пробел для строк, второе поле которых не содержит символ E

```
13. awk '$3 ~ /^Joel/{print $3 " is a nice guy."}' datafile
```

В результате работы программы будет выведено третье поле и строка is a nice guy. для исходных строк, третье поле которых начинается с подстроки Joel

```
14. awk '$8 ~ /[2-9][0-8]$/ {print $8}' datafile
```

В результате работы программы будет выведено восьмое поле для строк, восьмое поле которых заканчивается на две цифры, причем первая от 2 до 9, вторая – от 0 до 8

```
15. awk '$4 ~ /Chin$/{print "The price is $" $8 "."}' datafile
```

В результате работы программы будет выведена строка The price is \$, а за ней восьмое поле строк, четвертое поле которых заканчивается на подстроку Chin

```
16. awk '/TJ/{print $0}' datafile
```

В результате работы программы будут выведены строки, содержащие подстроку TJ