**Отчет по лабораторной работе № 20** по курсу “Фундаментальная информатика”

Студент группы M80-103Б-21 Субботина Мария Алексеевна, № по списку 19

Контакты e-mail, telegram: rejeverfaj@gmail.com, @MarySubb

Работа выполнена: «» сентября 2021г.

Преподаватель: каф. 806 Севастьянов Виктор Сергеевич

Отчет сдан « » \_\_\_\_\_\_\_\_\_20\_\_ г., итоговая оценка \_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Тема: стандартные утилиты UNIX для обработки файлов.**
2. **Цель работы: обработка файла с помощью утилит UNIX.**
3. **Задание** (*вариант №* )**: нет.**
4. **Оборудование** (студента):

Процессор AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx 2.10 GHz.

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

Операционная система семейства: *linux*, наименование: *ubuntu*, версия 20*.04.3 LTS*интерпретатор команд: *bash* версия 5.0.17

Система программирования -- версия --**,** редактор текстов *emacs* версия 3.24.14

Утилиты операционной системы --Прикладные системы и программы --

Местонахождение и имена файлов программ и данных на домашнем компьютере --

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

Утилиты:

1. cmp - побайтовое сравнение двух файлов.
2. comm - сравнивает отсортированные файлы построчно.
3. wc - печатает число строк, слов и байт для каждого файла.
4. dd - побайтовое копирование.
5. diff - сравнивает два файла построчно.
6. grep - используется для поиска строк, соответствующих строке в тексте или содержимому файлов.
7. join - объединяет строки двух упорядоченных текстовых файлов на основе наличия общего поля; сначала печатается поле, в котором были сопоставлены строки, за ним следуют другие поля из файла 1, а затем поля из файла 2 без поля сопоставления.
8. sort - сортировка.
9. tail - позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме.
10. tee - записывает вывод любой команды в один или несколько файлов.
11. tr - используется для замены, замещения или удаления символов из стандартного ввода, отправляя результат на стандартный вывод.
12. uniq - вывод или фильтрация повторяющихся строк в отсортированном файле.
13. od - по умолчанию преобразует входные данные в несколько форматов с восьмеричным форматом.
14. sum - используется для поиска контрольной суммы и подсчета блоков в файле.
15. cut - используется для вырезания текста в строках файла.
16. nroff - позволяет управлять процессом форматирования текстов.
17. vi/vim - текстовый редактор.
18. mc - текстовый файловый менеджер.
19. tar - используется для сжатия файлов и папок(архивирование). Упаковка и распаковка архивов tar.
20. gzip - создание, изменение, просмотр содержимого и распаковка архивов Gzip.
21. ed - строковый редактор.
22. awk - утилита для фильтрации текста на основе регулярных выражений и языка программирования AWK.
23. sed - потоковый редактор текста на основе регулярных выражений(работает по принципу замены).
24. bzip2 - используется для сжатия и распаковки файлов , часто используется в связке с tar.
25. head - отобразить первые 10 строк из файла.
26. iconv - используется для преобразования некоторого текста в одной кодировке в другую кодировку.
27. patch - добавляет файлы исправлений в исходный код или текстовые файлы.
28. md5 - проверка контрольной суммы( понять, что файл не был изменен при закачке).
29. du - отобразить занимаемое каждым файлом место на диске.
30. file - вывод типа файла.
31. touch - используется для создания новых пустых файлов, а также для обновления временных меток в уже существующих файлов и каталогов.
32. find - поиск файлов в файловой системе по разным условиям.
33. xargs - позволяет составлять команды на лету(конвейер, вывод предыдущей команды можно передать в аргументы следующей).
34. df - посмотреть общее доступное дисковое пространство в системе.
35. paste - объединить строки из файлов; объединяет несколько входных файлов для создания из них нового текстового файла с разделителями.
36. cpp - препроцессор языка C, он автоматически используется компилятором C для преобразования вашей программы перед компиляцией.
37. indent - помогает разметить исходный код пробелами и табуляциями в соответствии с теми или иными стандартами.
38. split - команда, копирующая файл и разбивающая его на отдельные файлы заданной длины.
39. mktemp - безопасно создает временный файл или каталог и выводит его имя.
40. **Сценарий выполнения работы** [план работы,первоначальный текст программы в черновике(можно на отдельном листе)итесты либо соображения по тестированию].

**Команды, используемые при обработке файла:**

awk '{print $0}' file1.txt : вывод текста файла

sed '/^$/d' file1.txt > tmpfile; mv tmpfile file1.txt : удаляем пустые строки

sed '/\\/d' file1.txt > tmpfile; mv tmpfile file1.txt : удаляем комментарии

awk '/^#\s\*include/' file1.txt : выводим б-ки

awk ' { if( $1 == "int" && $NF == "{" ) print; else print "---" }' file1.txt : выводим только строки с int, остальные заменяем на ---

awk '/^int .\*(.\*) {/, /^}/' file1.txt : выводим всё, что в int

sed -n '/^int/=' file1.txt : выводим номера строк с int в начале

awk '6 < NR' file1.txt : выводим все строки ниже 6-ой

sed 's/float/int/' file1.txt > tmpfile; mv tmpfile file1.txt : заменяем float на int

sed 's/100/10/' file1.txt > tmpfile; mv tmpfile file1.txt : заменяем 100 на 10

awk '$2 == "long"' file1.txt : выводим строки с long

sed -n '/long/=' file1.txt : выводим номера строк с long

sed 's/int y/int y, d, s, m = 1, l = 0/' file1.txt > tmpfile; mv tmpfile file1.txt : заменяем int y

sed '11,14d' file1.txt > tmpfile; mv tmpfile file1.txt : удаляем строки в заданном диапазоне

awk 'BEGIN {print "The File Contents:"}

> { printf "%-2d %s\n", NR, $0 }

> END {print "End of File", NR, "lines."}' file1.txt | tee file1.txt : нумеруем все строки файла, в начале и конце ставим заголовки, а также в конце подсчитываем количество строк файла

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

masha@Maha:~$ cmp file1.txt file2.txt

file1.txt file2.txt различаются: байт 1, строка 1

masha@Maha:~$ cmp -b file1.txt file2.txt

file1.txt file2.txt различаются: байт 1, строка 1 равен 167 w 165 u

masha@Maha:~$ wc file1.txt

8 16 84 file1.txt

masha@Maha:~$ wc file2.txt

8 8 49 file2.txt

masha@Maha:~$ dd if=file1.txt of=file2.txt

0+1 записей получено

0+1 записей отправлено

84 байта скопировано, 0,000359985 s, 233 kB/s

masha@Maha:~$ sort file1.txt

cv

dfg

hjk

lzx

masha@Maha:~$ sort file2.txt

ps

qwe

rty

uio

masha@Maha:~$ comm file1.txt file2.txt

cv

dfg

hjk

lzx

ps

qwe

rty

uio

masha@Maha:~$ diff file1.txt file2.txt

1,4c1,4

< cv

< dfg

< hjk

< lzx

---

> ps

> qwe

> rty

> uio

masha@Maha:~$ diff -s file1.txt file2.txt

1,4c1,4

< cv

< dfg

< hjk

< lzx

---

> ps

> qwe

> rty

> uio

masha@Maha:~$ diff -q file1.txt file2.txt

Файлы file1.txt и file2.txt различаются

masha@Maha:~$ grep -n 'u' file1.txt file2.txt

file2.txt:4:uio

masha@Maha:~$ tail -n 1 file1.txt file2.txt

==> file1.txt <==

lzx

==> file2.txt <==

uio

masha@Maha:~$ tee -a file1.txt

rtw

rtw

^C

masha@Maha:~$ cat file1.txt

cv

dfg

hjk

lzx

rtw

masha@Maha:~$ cut -c 1 file1.txt

c

d

h

l

r

masha@Maha:~$ sum file1.txt

03598 1

masha@Maha:~$ sed -n '1,3p' file1.txt

cv

dfg

hjk

masha@Maha:~$ od -b file1.txt

0000000 143 166 012 144 146 147 012 150 152 153 012 154 172 170 012 162

0000020 164 167 012

0000023

masha@Maha:~$ tar --totals -cvf achive.tar file2.txt file3.txt

file2.txt

file3.txt

Всего записано байт: 10240 (10KiB, 9,1MiB/s)

masha@Maha:~$ gzip -c file1.txt > 20.gz

masha@Maha:~$ ls

11.c df.txt jhb rty Документы

20 dlliza jhb11 sf Загрузки

20.gz fgh jhb11.c VirtualTuringMachine Изображения

...

masha@Maha:~$ cat file1.txt | awk '{print NF}'

1

1

1

1

1

masha@Maha:~$ cat file1.txt | awk '{print $NF}'

cv

dfg

hjk

lzx

rtw

masha@Maha:~$ cat file1.txt

cv

dfg

hjk

lzx

rtw

masha@Maha:~$ cat file1.txt | awk '{print $1}'

cv

dfg

hjk

lzx

rtw

masha@Maha:~$ join file3.txt file4

1 wefd ette

2 efc erff

3 eccc fcvd

4 sdvca dvca

5 sadcae sdca

masha@Maha:~$ tr c y

cciucuicbcccicu ccc

yyiuyuiybyyyiyu yyy

masha@Maha:~$ echo -e rrr\\ntttyt\\nrrr\\nrrr\\nyui\\niooi\\ndfggf\\nfgfh\\nhhh\\nhhh | uniq

rrr

tttyt

rrr

yui

iooi

dfggf

fgfh

hhh

masha@Maha:~$ echo -e rrr\\ntttyt\\nrrr\\nrrr\\nyui\\niooi\\ndfggf\\nfgfh\\nhhh\\nhhh | uniq -c

1 rrr

1 tttyt

2 rrr

1 yui

1 iooi

1 dfggf

1 fgfh

2 hhh

masha@Maha:~$ head file3.txt

c1 wefd

2 efc

3 eccc

4 sdvca

5 sadcae

cmasha@Maha:~$ head -c 10 file3.txt

c1 wefd

2 masha@Maha:~$ head -c -10 file3.txt

c1 wefd

2 efc

3 eccc

4 sdvca

masha@Maha:~$ head -n 3 file3.txt

c1 wefd

2 efc

3 eccc

masha@Maha:~$ df -a

Файл.система 1K-блоков Использовано Доступно Использовано% Cмонтировано в

sysfs 0 0 0 - /sys

proc 0 0 0 - /proc

udev 3495204 0 3495204 0% /dev

devpts 0 0 0 - /dev/pts

tmpfs 704968 1764 703204 1% /run

/dev/nvme0n1p8 28705700 10761900 16462584 40% /

securityfs 0 0 0 - /sys/kernel/security

tmpfs 3524832 0 3524832 0% /dev/shm

tmpfs 5120 4 5116 1% /run/lock

tmpfs 3524832 0 3524832 0% /sys/fs/cgroup

...

masha@Maha:~$ df -h -T file4

Файл.система Тип Размер Использовано Дост Использовано% Cмонтировано в

/dev/nvme0n1p9 ext4 72G 1,2G 68G 2% /home

masha@Maha:~$

masha@Maha:~$ ls

11.c df.txt jhb sf Изображения

20 dlliza jhb11 VirtualTuringMachine кер.c

20.1 fgh jhb11.c wc Музыка

20.gz file1.txt jhb.c 'Без имени 10'

...

masha@Maha:~$ split file4

masha@Maha:~$ ls

11.c df.txt jhb sf Загрузки

20 dlliza jhb11 VirtualTuringMachine Изображения

20.1 fgh jhb11.c wc кер.c

20.gz file1.txt jhb.c xaa Музыка

'A1(2).c' file2.txt lab12.c 'Без имени 10'

...

masha@Maha:~$ cat xaa

aaaa ddd cc addd

adbd

masha@Maha:~$ split -l 1 file4

masha@Maha:~$ ls

11.c df.txt jhb sf Документы

20 dlliza jhb11 VirtualTuringMachine Загрузки

20.1 fgh jhb11.c wc Изображения

20.gz file1.txt jhb.c xaa кер.c

'A1(2).c' file2.txt lab12.c xab Музыка

A1.c file3.txt lab20 'Без имени 10'

...

masha@Maha:~$ cat xaa

aaaa ddd cc addd

masha@Maha:~$ split -b 10 file4

masha@Maha:~$ ls

11.c df.txt jhb sf Видео

20 dlliza jhb11 VirtualTuringMachine Документы

20.1 fgh jhb11.c wc Загрузки

20.gz file1.txt jhb.c xaa Изображения

'A1(2).c' file2.txt lab12.c xab кер.c

A1.c file3.txt lab20 xac Музыка

A.c file4 lab9 'Без имени 10'

...

masha@Maha:~$ cat xaa

aaaa ddd c

masha@Maha:~$ paste -d '' file3.txt file4

aaaa bbbb ccaaaa ddd cc addd

adbd

masha@Maha:~$ paste -s file3.txt file4

aaaa bbbb cc

aaaa ddd cc adddadbd

masha@Maha:~$ du file4

4file4

masha@Maha:~$ du file3.txt file4

4file3.txt

4file4

masha@Maha:~$ mktemp

/tmp/tmp.xwFtGKOvrJ

masha@Maha:~$ mktemp -d

/tmp/tmp.oZrAiApcDE

masha@Maha:~$ mktemp -u

/tmp/tmp.GMYETNXS4g

masha@Maha:~$ touch 23.txt

masha@Maha:~$ ls

11.c fgh lab20 'Без имени 7'

20 file1.txt lab9 'Без имени 8'

20.1 file2.txt lab9.c 'Без имени 9'

20.gz file3.txt liza Видео

23.txt file4

...

masha@Maha:~$ touch 23{1..3}.txt

masha@Maha:~$ is

is: команда не найдена

masha@Maha:~$ ls

20 a.out id sf Документы

20.1 archive.tar jhb VirtualTuringMachine Загрузки

20.gz df.txt jhb11 wc Изображения

231.txt dlliza jhb11.c xaa кер.c

232.txt fgh jhb.c xab Музыка

233.txt file1.txt lab12.c xac Общедоступные

23.txt

masha@Maha:~$ find ./Документы

./Документы

./Документы/Без имени 5.odt

...

./Документы/Без имени 4.odt

masha@Maha:~$ ls ~/Документы | xargs -t -L1 echo

echo Без имени 1.odt

Без имени 1.odt

...

echo Без имени 5.odt

Без имени 5.odt

masha@Maha:~$ mc

masha@Maha:~$ ed file4

a

queyvc weucb

eicb

vihb

.

p

vihb

,p

aaaa ddd cc addd

adbd

queyvc weucb

eicb

vihb

f file4

file4

w

48

q

masha@Maha:~$ cat file4

aaaa ddd cc addd

adbd

queyvc weucb

eicb

Vihb

**Обработка файла**

masha@Maha:~$ awk '{print $0}' file1.txt

#include <stdio.h>

#include <math.h>

int abs(int a) {

if (a >= 0) {

return a;

}

\\ денег нет, но вы держитесь

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y;

long long int d;

\\ спааать...

long long int m = 1;

long long int l = 0;

long long float s;

int k = 0;

while (d > 0) {

d = d / 10;

n = n + 1;

}

f = n % 2;

\\ кофе - это жизнь!

s = s / 100;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

printf("%lld\n", s);

}

}

return 0;

}

masha@Maha:~$ sed '/^$/d' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ sed '/\\/d' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ awk '{print $0}' file1.txt

#include <stdio.h>

#include <math.h>

int abs(int a) {

if (a >= 0) {

return a;

}

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y;

long long int d;

long long int m = 1;

long long int l = 0;

long long float s;

int k = 0;

while (d > 0) {

d = d / 10;

n = n + 1;

}

f = n % 2;

s = s / 100;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

}

}

return 0;

}

masha@Maha:~$ awk '/^#\s\*include/' file1.txt

#include <stdio.h>

#include <math.h>

masha@Maha:~$ awk ' { if( $1 == "int" && $NF == "{" ) print; else print "---" }' file1.txt

---

---

int abs(int a) {

---

---

---

int main(void) {

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

masha@Maha:~$ awk '/^int .\*(.\*) {/, /^}/' file1.txt

int abs(int a) {

if (a >= 0) {

return a;

}

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y;

long long int d;

long long int m = 1;

long long int l = 0;

long long float s;

int k = 0;

while (d > 0) {

d = d / 10;

n = n + 1;

}

f = n % 2;

s = s / 100;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

}

}

return 0;

}

masha@Maha:~$ sed -n '/^int/=' file1.txt

3

7

masha@Maha:~$ awk '6 < NR' file1.txt

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y;

long long int d;

long long int m = 1;

long long int l = 0;

long long float s;

int k = 0;

while (d > 0) {

d = d / 10;

n = n + 1;

}

f = n % 2;

s = s / 100;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

}

}

return 0;

}

masha@Maha:~$ sed 's/float/int/' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ sed 's/100/10/' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ awk '{print $0}' file1.txt

#include <stdio.h>

#include <math.h>

int abs(int a) {

if (a >= 0) {

return a;

}

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y;

long long int d;

long long int m = 1;

long long int l = 0;

long long int s;

int k = 0;

while (d > 0) {

d = d / 10;

n = n + 1;

}

f = n % 2;

s = s / 10;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

}

}

return 0;

}

masha@Maha:~$ awk '$2 == "long"' file1.txt

long long int y;

long long int d;

long long int m = 1;

long long int l = 0;

long long int s;

masha@Maha:~$ sed -n '/long/=' file1.txt

10

11

12

13

14

masha@Maha:~$ sed 's/int y/int y, d, s, m = 1, l = 0/' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ sed '11,14d' file1.txt > tmpfile; mv tmpfile file1.txt

masha@Maha:~$ awk '{print $0}' file1.txt

#include <stdio.h>

#include <math.h>

int abs(int a) {

if (a >= 0) {

return a;

}

int main(void) {

int x;

while (scanf("%d", &x) != EOF) {

long long int y, d, s, m = 1, l = 0;

}

f = n % 2;

s = s / 10;

s = s \* m + l;

if (k == 1) {

s = s \* (-1);

}

}

}

return 0;

}

masha@Maha:~$ awk 'BEGIN {print "The File Contents:"}

> { printf "%-2d %s\n", NR, $0 }

> END {print "End of File", NR, "lines."}' file1.txt | tee file1.txt

The File Contents:

1 #include <stdio.h>

2 #include <math.h>

3 int abs(int a) {

4 if (a >= 0) {

5 return a;

6 }

7 int main(void) {

8 int x;

9 while (scanf("%d", &x) != EOF) {

10 long long int y, d, s, m = 1, l = 0;

11 }

12 f = n % 2;

13 s = s / 10;

14 s = s \* m + l;

15 if (k == 1) {

16 s = s \* (-1);

17 }

18 }

19 }

20 return 0;

21 }

End of File 21 lines.

1. **Дневник отладки** должен содержать дату и время сеансов отладки и основные события(ошибки в сценарии и программе,нестандартные ситуации) 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 | Действия по исправлению | Примечание |
| 1 | дом | 20.02 | \_ | \_ | \_ | \_ |

1. **Замечания автора** по существу работы: Нет.

**Выводы:** Я изучила и отработала стандартные утилиты UNIX для обработки данных. Из процесса выполнения поставленной задачи стоит отметить следующее: в UNIX есть множество простых утилит, благодаря которым многие сложные задачи обработки файлов можно выполнить путем последовательного выполнения этих программ. Более детально стоит рассмотреть команды awk и sed:Команда sed - это потоковый редактор текста, работающий по принципу замены. Его можно использовать для поиска, вставки, замены и удаления фрагментов в файле. Это очень гибкий инструмент, с помощью него очень удобно решать многие задачи редактирования файлов или фильтрации вывода. Команда awk - один из самых мощных инструментов для обработки и фильтрации текста. Это не просто утилита, а целый язык, разработанный для обработки данных. В отличие от sed, awk может помнить контекст, делать сравнения и многие другие вещи, которые могут делать другие языки программирования. Его можно использовать для быстрого и удобного форматирования текста.

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_