

Лабораторная работа №12

Архитектура ОС

Демидович. Н. М.

29 апреля 2023

Российский университет дружбы народов, Москва, Россия

Докладчик

- Демидович Никита Михайлович
- Студент группы НКАбд-01-22
- Кафедра прикладной информатики и теории вероятностей
- Российский университет дружбы народов
- 1132221550@pfur.ru
- <https://github.com/nikdem1>



Цели и задачи

Цель данной лабораторной работы - изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Задачи

Последовательность выполнения работы:

1. Написать командный файл, реализующий упрощённый механизм семафоров.
Командный файл должен в течение некоторого времени t_1 дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени $t_2 < t_1$, также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом). Запустить командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой (`> /dev/tty#`, где `#` — номер терминала куда перенаправляется вывод), в котором также запущен этот файл, но не фоновом, а в привилегированном режиме. Доработать программу так, чтобы имелась возможность взаимодействия трёх и более процессов.
2. 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

Теоретическое введение

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера. В операционных системах типа UNIX/Linux наиболее часто используются следующие реализации командных оболочек:

- оболочка Борна (Bourne shell или sh) — стандартная командная оболочка UNIX/Linux, содержащая базовый, но при этом полный набор функций;
- C-оболочка (или csh) — надстройка на оболочкой Борна, использующая C-подобный синтаксис команд с возможностью сохранения истории выполнения команд;
- оболочка Корна (или ksh) — напоминает оболочку C, но операторы управления программой совместимы с операторами оболочки Борна;
- BASH — сокращение от Bourne Again Shell (опять оболочка Борна), в основе своей совмещает свойства оболочек C и Корна (разработка компании Free Software

Выполнение лабораторной работы

Далее описан ход выполнения данной лабораторной работы.

На первом этапе выполнения работы я создал отдельную директорию для дальнейшей работы, командный файл programm1.sh и приступил к написанию командного файла, реализующего упрощённый механизм семафоров.

```
#!/bin/bash

lockfile="./lock.file"
exec {fn}>$lockfile

while test -f "$lockfile"
do
if flock -n ${fn}
then
    echo "File is blocked"
    sleep 5
    echo "File is unlocked"
    flock -u ${fn}
else
```

Чтобы реализовать команду `man` с помощью командного файла, я изучил содержимое каталога `/usr/share/man/man1`.

После этого я создал исполняемый файл для второй программы.

```
#!/bin/bash

a=$1
if test -f "/usr/share/man/man1/${a}.1.gz"
then less /usr/share/man/man1/${a}.1.gz
else
echo "There is no such command"
fi
```


И на финальном этапе выполнения работы я написал командный файл programm3.sh, генерирующий случайную последовательность букв латинского алфавита.

Мсполняемый файл programm3.sh

```
#!/bin/bash
```

```
a=$1
```

```
for ((i=0; i<$a; i++))
```

```
do
```

```
  ((char=$RANDOM%26+1))
```

```
  case $char in
```

```
    1) echo -n a;; 2) echo -n b;; 3) echo -n c;; 4) echo -n d;; 5) echo -n  
    7) echo -n g;; 8) echo -n h;; 9) echo -n i;; 10) echo -n j;; 11) echo -  
    13) echo -n m;; 14) echo -n n;; 15) echo -n o;; 16) echo -n p;; 17) echo  
    19) echo -n t;; 20) echo -n q;; 21) echo -n u;; 22) echo -n v;;  
    23) echo -n w;; 24) echo -n x;; 25) echo -n y;; 26) echo -n z;;
```

```
  esac
```

Результаты

В результате выполнения данной лабораторной работы я изучил основы программирования в оболочке ОС UNIX и научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Список источников

Лабораторная работа №12 (Архитектура ОС).