

ПРОГРАММНЫЕ КАНАЛЫ

Системные вызовы и библиотеки Unix SVR4

Иртегов Д.В.

ФФ/ФИТ НГУ

Электронный лекционный курс подготовлен в рамках реализации

Программы развития НИУ-НГУ на 2009-2018 г.г.

ЦЕЛИ РАЗДЕЛА

- Описать, как программные каналы используются для межпроцессного взаимодействия
- Определить два типа каналов
- Научить использовать системные вызовы, работающие с программными каналами
- Научить использовать стандартные библиотечные функции, работающие с программными каналами

Программные каналы (трубы)

- предназначены для передачи данных между процессами
- могут передавать неограниченное количество данных
- автоматическая синхронизация `open(2)`
- автоматическая блокировка при заполнении и исчерпании
- типы каналов:
 - каналы (неименованные каналы)
 - именованные каналы

pipe(2)

ИСПОЛЬЗОВАНИЕ

```
#include <unistd.h>
```

```
int pipe (int fildes[2]);
```

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

0 - успех

-1 - неуспех, и errno установлена.

ОСОБЕННОСТИ СИСТЕМНЫХ ВЫЗОВОВ

`open(2)` - не нужен

`close(2)` - когда доступ к каналу закончен

`read(2)` - обычно блокируется, если канал пуст

`write(2)` - обычно блокируется, если канал
заполнен

`lseek(2)` - не допустим

`dup(2)` - используется для перенаправления
ввода/вывода в канал

`fcntl(2)` - может установить режим ввода/вывода
без задержек

popen(3S)

ИСПОЛЬЗОВАНИЕ

```
#include <stdio.h>
```

```
FILE *popen (const char *command,  
             const char *type);
```

```
int pclose (FILE *stream);
```

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

успех –

popen: указатель на файл

pclose: статус подпроцесса

неуспех –

popen: нулевой указатель

pclose: -1

p2open(3G)

ИСПОЛЬЗОВАНИЕ

```
cc [flag ...] file ... -lgen [library ...]
```

```
#include <libgen.h>
```

```
int p2open (const char *command,  
            FILE *fp[2]);
```

```
int p2close (FILE *fp[2]);
```

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

успех —

p2open: pid подпроцесса

p2close: статус подпроцесса

неуспех —

p2open: 0

p2close: -1

mknod(2)

ИСПОЛЬЗОВАНИЕ

```
#include <sys/types.h>
```

```
#include <sys/stat.h>
```

```
int mknod (const char *path, mode_t mode, dev_t dev);
```

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

0 - успех

-1 - неуспех, и errno установлена

Для создания именованной трубы,

- mode = S_IFIFO | access_rights
- dev = 0

Именованные трубы

`open(2)` – блокируется, пока не будет открыт другой конец канала

`close(2)` - когда доступ к каналу закончен

`read(2)` - обычно блокируется, если канал пуст

`write(2)` - обычно блокируется, если канал заполнен

`lseek(2)` - не допустим

`dup(2)` - используется для перенаправления ввода/вывода в канал

`fcntl(2)` - может установить режим ввода/вывода без задержек