

Multiplexação de E/S: Funções `select` e `poll`

MC 833 – Programação em Redes de
Computadores
Instituto de Computação – UNICAMP

Juliana Freitag Borin

Conteúdo do capítulo 6 do livro da bibliografia principal.

Relembrando: select

```
int select (int maxfdp1, fd_set *readset, fd_set
*writeset, fd_set *exceptset, const struct
timeval *timeout);
```

Quando um descritor está pronto?

Para leitura:

- quantidade de bytes no buffer de leitura é maior ou igual a um limitante inferior (*low-water mark* no livro). A opção do socket `SO_RCVLOWAT` permite atribuir o valor do limitante inferior;
- Uma direção da conexão está fechada. Leitura não irá bloquear e retornará zero;
- O socket é um *listening socket* e o número de conexões completadas não é nulo;
- Condição de erro pendente, retorna -1 em `errno`. Condições de erro podem ser verificadas e anuladas com a opção `getsockopt`.

Quando um descritor está pronto?

Para escrita:

- Quantidade de espaço disponível (em bytes) no buffer de escrita é maior que limite inferior e i) socket está conectado ou ii) não requer conexão (UDP). O limitante pode ser especificado pela opção `SO_SNDLOWAT`;
- A direção na conexão está fechada. Retorna `SIGPIPE`;
- se o socket estiver processando um `connect` (não bloqueante) e a conexão for estabelecida com sucesso ou se a conexão falhar;
- Condição de erro pendente.

Quando um descritor está pronto?

Condição de exceção existente:

- dados fora da faixa;
- se o socket estiver processando `connect` e o `connect` falhar.

Função poll

```
#include <poll.h>
```

```
int poll(struct pollfd *fdarray, unsigned long nfd, int timeout);
```

```
    /*retorna # de descritores prontos, 0 se timeout, -1 se erro.*/
```

```
struct pollfd{
```

```
    int fd; /*descritor a ser verificado*/
```

```
    short events; /*eventos de interesse para fd (bitmap) */
```

```
    short revents; /*eventos que ocorreram para fd (bitmap) */
```

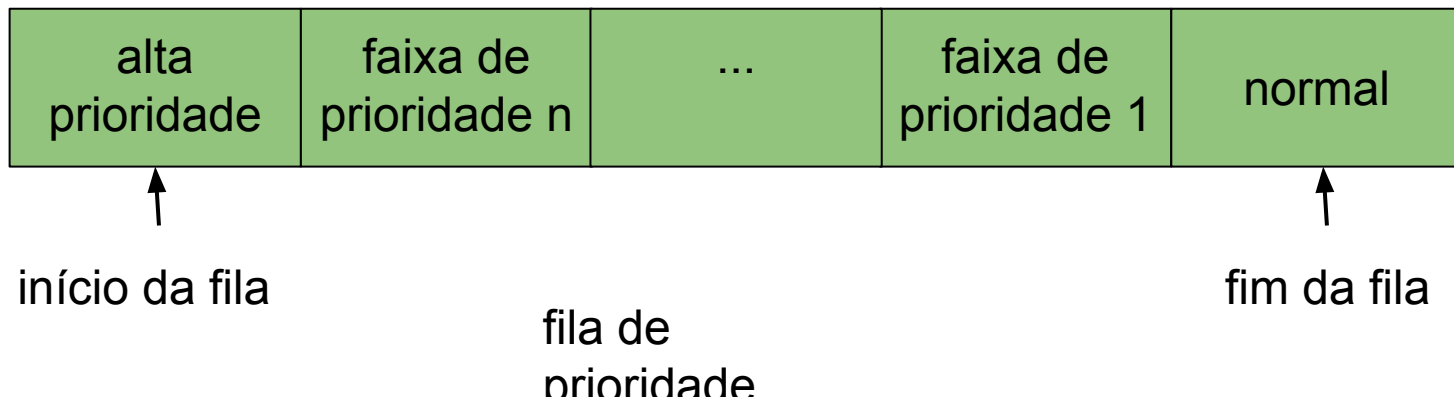
Função poll

Constantes usadas para especificar os eventos de interesse (**events**) e verificar os eventos que ocorreram (**revents**)

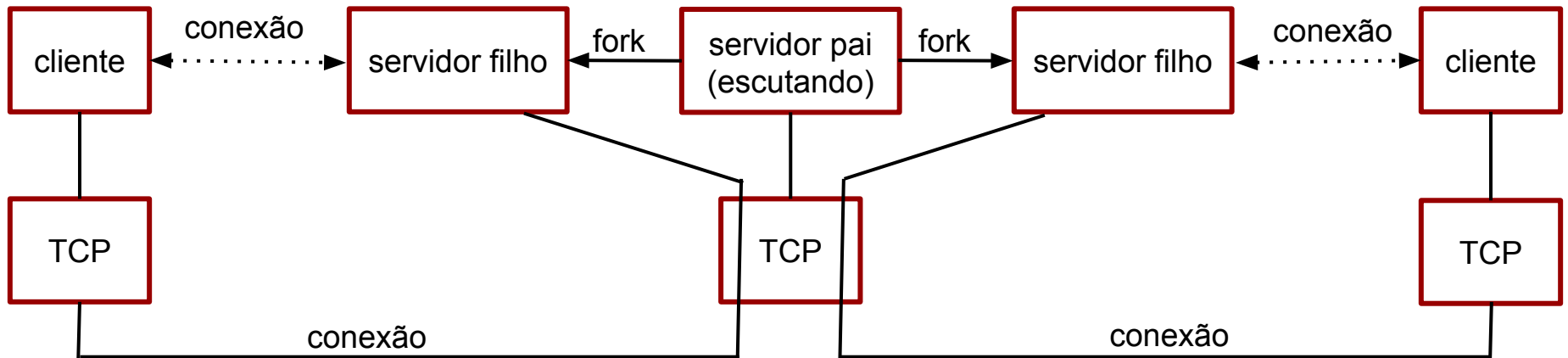
Constant	Input to events ?	Result from revents ?	Description
POLLIN	X	X	Normal or priority band data can be read
POLLRDNORM	X	X	Normal data can be read
POLLRDBAND	X	X	Priority band data can be read
POLLPRI	X	X	High-priority data can be read
POLLOUT	X	X	Normal data can be written
POLLWRNORM	X	X	Normal data can be written
POLLWRBAND	X	X	Priority band data can be written
POLLERR		X	Error has occurred
POLLHUP		X	Hangup has occurred
POLLNVAL		X	Descriptor is not an open file

Função poll

- Três tipos de dados:
 - ✓ normal;
 - ✓ faixa prioritária (*priority band*);
 - ✓ alta prioridade.
- Dados TCP e UDP são normais.
- TCP fora de faixa (*out-of-band*) é prioritário.
- Nova conexão em socket listening pode ser considerada normal ou prioritária.

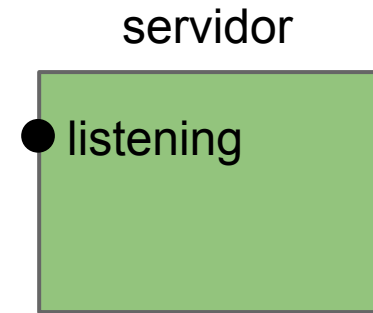


fork X multiplexação

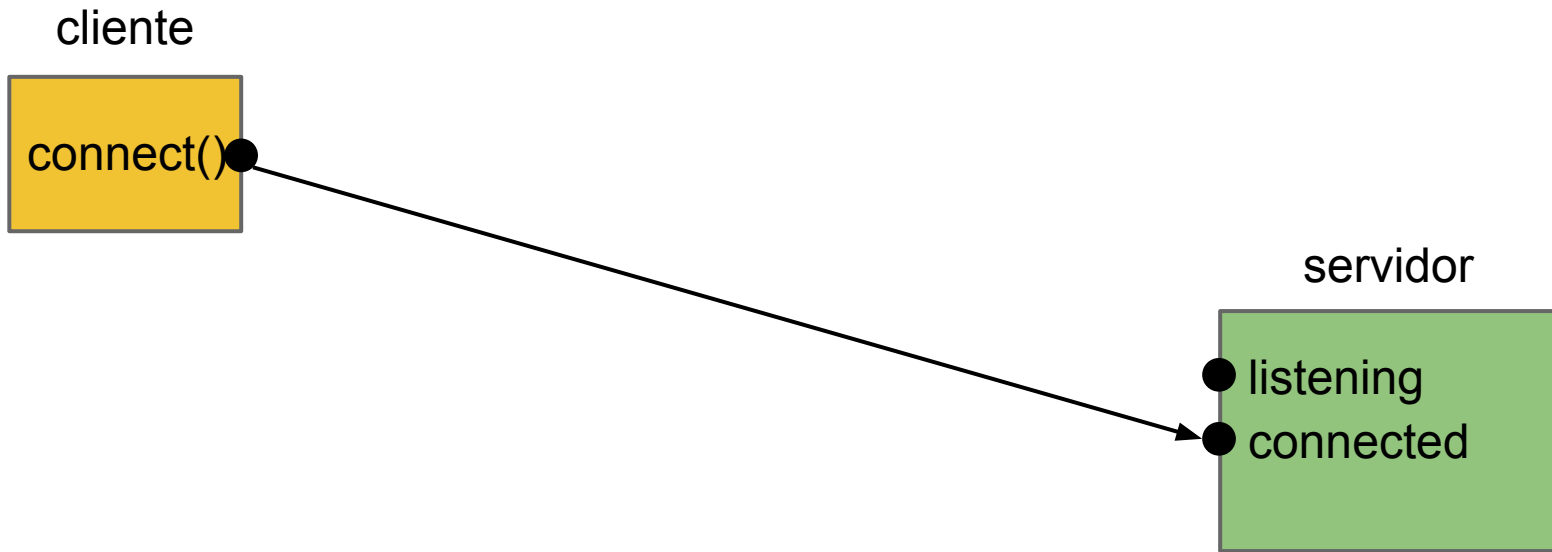


Cliente/Servidor TCP com dois clientes - usando `fork`

fork X multiplexação



fork X multiplexação



fork X multiplexação

