

Sistemas Operativos*Exame¹*

6 de Julho de 2010

Duração: 2h00m

I

É vulgar encontrar nas páginas web um conjunto de banners, botões, separadores, pequenas imagens, etc., que tipicamente se encontram guardadas num sistema de ficheiros. Suponha que esse conjunto é muito grande e que tem havido queixas de alguma lentidão no carregamento das páginas web.

1 Identifique algumas das causas prováveis para a referida lentidão. Como é óbvio, já terá feitos testes de desempenho com um cliente na rede local de alta velocidade, pelo que o problema estará algures no servidor web.

2 Admita que o serviço web corre sobre um sistema de memória virtual paginada. Vê alguma vantagem em carregar os ficheiros para memória virtual em vez de os aceder através da API habitual (open/read/close)? Justifique.

3 Sugere alguma alternativa que lhe pareça ter desempenho superior às indicadas? Explique a sua proposta e analise os seus aspectos positivos e negativos.

II

Pretende-se desenvolver um programa que archive o log produzido no stdout por um comando, passado como argumento, em `/var/log/nome_do_comando`. Para poder efectuar o processamento do log é necessário de 10 em 10 minutos fazer a rotação deste utilizando o programa `logrotate`, que recebe como argumento o ficheiro em que está a ser efectuado o log. Enquanto o `logrotate` estiver a correr deverá ser parada a escrita no ficheiro. Desenvolva o programa `MyLogRotate` que implemente as funcionalidades pretendidas: arrancar o comando, garantir que o log é escrito em `/var/log/nome_do_comando` e fazer o rotate do log de 10 em 10 minutos, parando a escrita deste durante a operação de rotate.

III

Codifique um programa `radio` que recebe como argumentos o nome de vários comandos. O programa deve correr os comandos e mostrar o output destes em alternância. O output de um dado programa é mostrado, antes de passar ao seguinte, até aparecer uma linha contendo unicamente a palavra "OVER". Depois do último passa ao primeiro e a alternância deve continuar assim sucessivamente entre os que ainda não tenham terminado.

*Protótipos de algumas funções e chamadas ao sistema relevantes***Processos**

- `pid_t fork(void);`
- `void exit(int status);`
- `pid_t wait(int *status);`
- `pid_t waitpid(pid_t pid, int *status, int options);`
- `WIFEXITED(status);`
- `WEXITSTATUS(status);`
- `int execlp(const char *file, const char *arg, ...);`
- `int execvp(const char *file, char *const argv[]);`
- `int execve(const char *file, char *const argv[], char *const envp[]);`

Sinais

- `void (*signal(int signum, void (*handler)(int)))(int);`
- `int kill(pid_t pid, int signum);`
- `int alarm(int seconds);`
- `int pause(void);`

Sistema de Ficheiros

- `int open(const char *pathname, int flags, mode_t mode);`
- `int creat(const char *pathname, mode_t mode);`
- `int close(int fd);`
- `int read(int fd, void *buf, size_t count);`
- `int write(int fd, const void *buf, size_t count);`
- `int pipe(int filedes[2]);`
- `int dup(int oldfd);`
- `int dup2(int oldfd, int newfd);`

¹Cotação — 8+6+6