

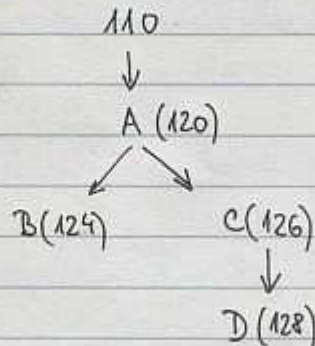


OPERÁČNÍ SYSTÉMY II.

CM

2006-12-12

1. A (PID=120, PPID=110)
B (PID=124, PPID=120)
C (PID=126, PPID=120)
D (PID=128, PPID=126)



a) zruším A:

	PID	PPID
A	—	—
B	124	1
C	126	1
D	128	126

zruším B - nic se nestane

zruším C - a D se změní PPID na 1

- b) číslo skupiny procesů - odpovídá tomu, kdo má PID stejný, jako číslo procesu $\Rightarrow 120$
A volá službu `exit()` - signál `SIGHUP` bude poslán všem, kdo jsou v jeho skupině a ty skončí
D volá službu `exit()` - signál `SIGCHLD` bude poslán rodiči (c)

2. Nejprve se provede `fork`, volá se `getenv`.

Pracovní prázdný pointer - služba není definována.

`exec` - spustí s parametry.Dětský proces má `exit(1)` jen když se nepodaří spustit `ls`, je-li OK, mácí nálev.

- V systému existují procesy A (PID=120, PPID=110), B (PID=124, PPID=120), C (PID=126, PPID=120), D (PID=128, PPID=126).
 - Jak se změní PID a PPID procesů pokud zruším proces:
 - A, b) B, c) C.
 - Předpokládejte, že proces A je vedoucím skupiny procesů s přiřazeným řídicím terminálem a B, C, D jsou členy této skupiny procesů.
 - Jaké je číslo skupiny procesů?
 - Pokud proces A volá službu `exit()`, kterému z procesů zašle OS signál `SIGHUP`.
 - Pokud proces D volá službu `exit()`, kterému z procesů zašle OS signál `SIGCHLD`.



2. Napište program, který vypíše spuštěním programu **ls -al** adresář zadaný hodnotou proměnné prostředí ADR. Pokud proměnná ADR nebude definována, vypíše obsah aktuálního adresáře.

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int pid,status;
    char *padr;
    if ((pid = fork()) == 0){
        if ((padr=getenv("ADR")) == NULL)
            padr=".";
        execl("/bin/ls","ls","-al",padr,NULL);
        exit(1);
    }
    wait(&status);
    if(status){
        printf("ls nebyl spusten nebo skoncil s chybou\n");
        exit(1);
    }
    else
        printf("program ls skoncil O.K. \n");
        exit(0);
}
```

3. Napište program, který bude každých 5 sekund vypisovat na obrazovku řetězec znaků. Program napište tak, aby pokud poběží na popředí nebylo možné jej zrušit stisknutím kláves CTRL+C. Vyzkoušejte zda po příchodu signálu se nastavení ošetření signálu ruší nebo ne. Pozměňte program tak, aby ignoroval pouze jedno stisknutí kláves CTRL+C.

```
#include <signal.h>
#include <stdio.h>
void handler(sig)
int sig;
{ if (sig == 2)
    printf("signalem 2 mne nelze zrusit\n");
    signal(2,handler);
    signal(14,handler);
}
main()
{ int i;
  signal(2,handler);
  signal(14,handler);
  for(i=0;i<100;i++){
      alarm(5);
      pause();
      printf("hello\n");
  }
}
```