SIGHUP 1 A 终端挂起或者控制进程终止

SIGINT 2 A 键盘中断（如break键被按下）

SIGQUIT 3 C 键盘的退出键被按下

SIGILL 4 C 非法指令

SIGABRT 6 C 由abort(3)发出的退出指令

SIGFPE 8 C 浮点异常

SIGKILL 9 AEF Kill信号

SIGSEGV 11 C 无效的内存引用

SIGPIPE 13 A 管道破裂: 写一个没有读端口的管道

SIGALRM 14 A 由alarm(2)发出的信号

SIGTERM 15 A 终止信号

SIGUSR1 30,10,16 A 用户自定义信号1

SIGUSR2 31,12,17 A 用户自定义信号2

SIGCHLD 20,17,18 B 子进程结束信号

SIGCONT 19,18,25 进程继续（曾被停止的进程）

SIGSTOP 17,19,23 DEF 终止进程

SIGTSTP 18,20,24 D 控制终端（tty）上按下停止键

SIGTTIN 21,21,26 D 后台进程企图从控制终端读

SIGTTOU 22,22,27 D 后台进程企图从控制终端写

void (\*signal(int sig\_name,void (\*f)(int)))(int)

#include<signal.h>

相当于

typedef void (\*TYPE)(int)

TYPE signal(int sig\_name,TYPE callback)

int kill(pid\_t pid,int sig\_name)

int raise(int sig\_name)

#include<sys/types.h>

#include<signal.h>

unsigned int alarm(unsigned int seconds)

int pause() 进程挂起直至捕捉到一个信号

#include<unistd.h>