<https://blog.csdn.net/fly__chen/article/details/53101576>

Linux系统下错误处理方法：使用errno变量。头文件： errno.h

在系统出现错误时内核把错误写入到变量errno中。用户的应用程序可以从这个变量中获取错误的原因。

注意在系统调用返回失败时，必须紧接着引用errno变量，以避免程序中的下一条系统调用时覆盖errno的值。

errno的值可以格式化使出，使用错误处理函数 : perror strerror。

perror和strerror函数都是用来打印错误提示信息的。

perror：将错误信息出处到终端。   
strerror：将错误信息输出到指定的缓冲区

它们的原型分别是：

perror:

#include <stdio.h>

void perror(const char \*s);

它先打印s指向的字符串，然后输出当前errno值所对应的错误提示信息，例如当前errno若为12，

调用perror("ABC")，会输出"ABC: Cannot allocate memory"。

测试程序：

#include <stdio.h>

#include <unistd.h>

int main(void)

{

int fd = 10;

int ret;

ret = close(fd);

if(ret == -1)

perror("close error");

return 0;

}

测试结果：

close error: Bad file descriptor

strerror:

#include <string.h>

char \*strerror(int errnum);

它返回errnum的值所对应的错误提示信息，例如errnum等于12的话，

它就会返回"Cannot allocate memory"。

测试程序：

#include <stdio.h>

#include <string.h>

#include <unistd.h>

#include <errno.h>

int main(void)

{

int fd = 10;

int ret;

ret = close(fd);

if(ret == -1)

fprintf(stderr, "close error with msg is: %s\n",strerror(errno));

return 0;

}

测试结果：

close error with msg is: Bad file descriptor