NAME

sysexits - preferable exit codes for programs

SYNOPSIS

#include <sysexits.h>

DESCRIPTION

According to style(9), it is not a good practice to call exit(3) with arbitrary values to indicate a failure condition when ending a program. Instead, the pre-defined exit codes from **sysexits** should be used, so the caller of the process can get a rough estimation about the failure class without looking up the source code.

The successful exit is always indicated by a status of 0, or **EX_OK**. Error numbers begin at **EX__BASE** to reduce the possibility of clashing with other exit statuses that random programs may already return. The meaning of the codes is approximately as follows:

EX_USAGE (64)	The command was used incorrectly, e.g., with the wrong number of arguments, a bad flag, a bad syntax in a parameter, or whatever.
EX_DATAERR (65)	The input data was incorrect in some way. This should only be used for user's data and not system files.
EX_NOINPUT (66)	An input file (not a system file) did not exist or was not readable. This could also include errors like "No message" to a mailer (if it cared to catch it).
EX_NOUSER (67)	The user specified did not exist. This might be used for mail addresses or remote logins.
EX_NOHOST (68)	The host specified did not exist. This is used in mail addresses or network requests.
EX_UNAVAILABLE (69)	A service is unavailable. This can occur if a support program or file does not exist. This can also be used as a catchall message when something you wanted to do does not work, but you do not know why.
EX_SOFTWARE (70)	An internal software error has been detected. This should be limited to non-operating system related errors as possible.
EX_OSERR (71)	An operating system error has been detected. This is intended to be

used for such things as "cannot fork", "cannot create pipe", or the like. It includes things like getuid returning a user that does not exist in the passwd file.

EX_OSFILE (72) Some system file (e.g., */etc/passwd*, */var/run/utx.active*, etc.) does not exist, cannot be opened, or has some sort of error (e.g., syntax error).

EX_CANTCREAT (73) A (user specified) output file cannot be created.

EX_IOERR (74) An error occurred while doing I/O on some file.

EX_TEMPFAIL (75) Temporary failure, indicating something that is not really an error. In sendmail, this means that a mailer (e.g.) could not create a connection, and the request should be reattempted later.

EX_PROTOCOL (76) The remote system returned something that was "not possible" during a protocol exchange.

EX_NOPERM (77) You did not have sufficient permission to perform the operation. This is not intended for file system problems, which should use **EX_NOINPUT** or **EX_CANTCREAT**, but rather for higher level permissions.

EX_CONFIG (78) Something was found in an unconfigured or misconfigured state.

The numerical values corresponding to the symbolical ones are given in parenthesis for easy reference.

SEE ALSO

err(3), exit(3), style(9)

HISTORY

The **sysexits** file appeared somewhere after 4.3BSD.

AUTHORS

This manual page was written by Jörg Wunsch after the comments in <*sysexits.h*>.

BUGS

The choice of an appropriate exit value is often ambiguous.