NAME

nm − list symbols from object files

SYNOPSIS

nm [**−a**|**−−debug−syms**] [**−g**|**−−extern−only**]
[**−B**] [**−C**|**−−demangle**[=*style*]] [**−D**|**−−dynamic**]
[**−S**|**−−print−size**] [**−s**|**−−print−armap**]
[**−A**|**−o**|**−−print−file−name**]
[**−n**|**−v**|**−−numeric−sort**] [**−p**|**−−no−sort**]
[**−r**|**−−reverse−sort**] [**−−size−sort**] [**−u**|**−−undefined−only**]
[**−t** *radix*|**−−radix=***radix*] [**−P**|**−−portability**]
[**−−target=***bfdname*] [**−f***format*|**−−format=***format*]
[**−−defined−only**] [**−l**|**−−line−numbers**] [**−−no−demangle**]
[**−V**|**−−version**] [**−X 32_64**] [**−−help**] [*objfile*...]

DESCRIPTION

GNU **nm** lists the symbols from object files *objfile*.... If no object files are listed as arguments, **nm** assumes the file *a.out*.

For each symbol, **nm** shows:

- The symbol value, in the radix selected by options (see below), or hexadecimal by default.
- The symbol type. At least the following types are used; others are, as well, depending on the object file format. If lowercase, the symbol is local; if uppercase, the symbol is global (external).

A The symbol's value is absolute, and will not be changed by further linking.

B The symbol is in the uninitialized data section (known as BSS).

C The symbol is common. Common symbols are uninitialized data. When linking, multiple common symbols may appear with the same name. If the symbol is defined anywhere, the common symbols are treated as undefined references.

D The symbol is in the initialized data section.

G The symbol is in an initialized data section for small objects. Some object file formats permit more efficient access to small data objects, such as a global int variable as opposed to a large global array.

I The symbol is an indirect reference to another symbol. This is a GNU extension to the a.out object file format which is rarely used.

N The symbol is a debugging symbol.

R The symbol is in a read only data section.

S The symbol is in an uninitialized data section for small objects.

T The symbol is in the text (code) section.

U The symbol is undefined.

V The symbol is a weak object. When a weak defined symbol is linked with a normal defined symbol, the normal defined symbol is used with no error. When a weak undefined symbol is linked and the symbol is not defined, the value of the weak symbol becomes zero with no error.

W The symbol is a weak symbol that has not been specifically tagged as a weak object symbol. When a weak defined symbol is linked with a normal defined symbol, the normal defined symbol is used with no error. When a weak undefined symbol is linked and the symbol is not defined, the value of the weak symbol becomes zero with no error.

− The symbol is a stabs symbol in an a.out object file. In this case, the next values printed are the stabs other field, the stabs desc field, and the stab type. Stabs symbols are used to hold debugging information.

? The symbol type is unknown, or object file format specific.

- The symbol name.

OPTIONS

The long and short forms of options, shown here as alternatives, are equivalent.

−A
−o
−−print−file−name

Precede each symbol by the name of the input file (or archive member) in which it was found, rather than identifying the input file once only, before all of its symbols.

−a
−−debug−syms

Display all symbols, even debugger-only symbols; normally these are not listed.

−B The same as **−−format=bsd** (for compatibility with the MIPS **nm**).

−C
−−demangle[=*style***]**

Decode (*demangle*) low-level symbol names into user-level names. Besides removing any initial underscore prepended by the system, this makes C++ function names readable. Different compilers have different mangling styles. The optional demangling style argument can be used to choose an appropriate demangling style for your compiler.

−−no−demangle

Do not demangle low-level symbol names. This is the default.

−D
−−dynamic

Display the dynamic symbols rather than the normal symbols. This is only meaningful for dynamic objects, such as certain types of shared libraries.

−f *format*
−−format=*format*

Use the output format *format*, which can be `bsd`, `sysv`, or `posix`. The default is `bsd`. Only the first character of *format* is significant; it can be either upper or lower case.

−g
−−extern−only

Display only external symbols.

−l
−−line−numbers

For each symbol, use debugging information to try to find a filename and line number. For a defined symbol, look for the line number of the address of the symbol. For an undefined symbol, look for the line number of a relocation entry which refers to the symbol. If line number information can be found, print it after the other symbol information.

−n
−v
−−numeric−sort

Sort symbols numerically by their addresses, rather than alphabetically by their names.

−p
−−no−sort

Do not bother to sort the symbols in any order; print them in the order encountered.

−P
−−portability

Use the POSIX.2 standard output format instead of the default format. Equivalent to **−f posix**.

−S
−−print−size
Print size, not the value, of defined symbols for the `bsd` output format.

−s
−−print−armap
When listing symbols from archive members, include the index: a mapping (stored in the archive by **ar** or **ranlib**) of which modules contain definitions for which names.

−r
−−reverse−sort
Reverse the order of the sort (whether numeric or alphabetic); let the last come first.

−−size−sort
Sort symbols by size. The size is computed as the difference between the value of the symbol and the value of the symbol with the next higher value. If the `bsd` output format is used the size of the symbol is printed, rather than the value, and **−S** must be used in order both size and value to be printed.

−t *radix*
−−radix=*radix*
Use *radix* as the radix for printing the symbol values. It must be **d** for decimal, **o** for octal, or **x** for hexadecimal.

−−target=*bfdname*
Specify an object code format other than your system's default format.

−u
−−undefined−only
Display only undefined symbols (those external to each object file).

−−defined−only
Display only defined symbols for each object file.

−V
−−version
Show the version number of **nm** and exit.

−X This option is ignored for compatibility with the AIX version of **nm**. It takes one parameter which must be the string **32_64**. The default mode of AIX **nm** corresponds to **−X 32**, which is not supported by GNU **nm**.

−−help
Show a summary of the options to **nm** and exit.

SEE ALSO

ar (1), *objdump* (1), *ranlib* (1), and the Info entries for *binutils*.

COPYRIGHT