

TABLE B-11. test operators

The test Command

General Format:

```
test condition
or
[ condition ]
```

The shell evaluates *condition* and if the result of the evaluation is *TRUE*, returns a zero exit status. If the result of the evaluation is *FALSE*, then a nonzero exit status is returned. If the format *[condition]* is used, then a space must appear immediately after the *[* and before the *]*.

condition is composed of one or more operators as shown in Table B-11.

The *-a* operator has higher precedence than the *-o* operator. In any case, parentheses can be used to group subexpressions. Just remember that the parentheses are significant to the shell and so must be quoted. Operators and operands (including parentheses) must be delimited by one or more spaces so that *test* sees them as separate arguments.

test is often used to test conditions in an *if*, *while*, or *until* command.

Examples

```
# see if perms is executable
```

```
if test -x /etc/perms
then
    ...
fi
```

```
# see if it's a directory or a normal file that's readable
```

```
if [ -d $file -o \( -f $file -a -r $file \) ]
then
    ...
fi
```

The Korn shell *test* command allows any arithmetic expression with the integer comparison operators (*-eq*, *-ge*, *-gt*, *-le*, *-lt*, and *-ne*).

Operator	Returns TRUE (zero exit status) if
<i>File Operators</i>	
<i>-b file</i>	<i>file</i> is a block special file
<i>-c file</i>	<i>file</i> is a character special file
<i>-d file</i>	<i>file</i> is a directory
<i>-f file</i>	<i>file</i> is an ordinary file
<i>-g file</i>	<i>file</i> has its set group id (SGID) bit set
<i>-G file</i>	<i>file's</i> group is the effective group id
<i>-k file</i>	<i>file</i> has its sticky bit set
<i>-L file</i>	<i>file</i> is a symbolic link (link across file systems)
<i>-O file</i>	<i>file's</i> owner is the effective user id
<i>-p file</i>	<i>file</i> is a named pipe
<i>-r file</i>	<i>file</i> is readable by the process
<i>-s file</i>	<i>file</i> has nonzero length
<i>-S file</i>	<i>file</i> is a socket
<i>-t fd</i>	<i>fd</i> is open file descriptor associated with a terminal (1 is default)
<i>-u file</i>	<i>file</i> has its set user id (SUID) bit set
<i>-w file</i>	<i>file</i> is writable by the process
<i>-x file</i>	<i>file</i> is executable
<i>file1 -ef file2</i>	<i>file1</i> and <i>file2</i> are linked
<i>file1 -nt file2</i>	<i>file1</i> is newer than <i>file2</i>
<i>file1 -ot file2</i>	<i>file1</i> is older than <i>file2</i>
<i>String Operators</i>	
<i>string</i>	<i>string</i> is not null
<i>-n string</i>	<i>string</i> is not null (<i>string</i> must be seen by <i>test</i>)
<i>-z string</i>	<i>string</i> is null (<i>string</i> must be seen by <i>test</i>)
<i>string₁ = string₂</i>	<i>string₁</i> is identical to <i>string₂</i>
<i>string₁ != string₂</i>	<i>string₁</i> is not identical to <i>string₂</i>
<i>Integer Comparison Operators</i>	
<i>int₁ -eq int₂</i>	<i>int₁</i> is equal to <i>int₂</i>
<i>int₁ -ge int₂</i>	<i>int₁</i> is greater than or equal to <i>int₂</i>
<i>int₁ -gt int₂</i>	<i>int₁</i> is greater than <i>int₂</i>
<i>int₁ -le int₂</i>	<i>int₁</i> is less than or equal to <i>int₂</i>
<i>int₁ -lt int₂</i>	<i>int₁</i> is less than <i>int₂</i>
<i>int₁ -ne int₂</i>	<i>int₁</i> is not equal to <i>int₂</i>
<i>Boolean Operators</i>	
<i>! expr</i>	<i>expr</i> is FALSE; otherwise returns TRUE
<i>expr₁ -a expr₂</i>	<i>expr₁</i> is TRUE and <i>expr₂</i> is TRUE
<i>expr₁ -o expr₂</i>	<i>expr₁</i> is TRUE or <i>expr₂</i> is TRUE