Squeezing characters with tr

The `tr` command is very helpful in many text-processing contexts. Repeated continuous characters should be squeezed to a single character in many circumstances. Squeezing of whitespace is a frequently occurring task.

`tr` provides the `-s` option to squeeze repeating characters from the input. It can be performed as follows:

```
$ echo "GNU is       not     UNIX. Recursive   right ?" | tr -s ' '
GNU is not UNIX. Recursive right ?
# tr -s '[set]'
```

Let's use `tr` in a tricky way to add a given list of numbers from a file as follows:

```
$ cat sum.txt
1
2
3
4
5

$ cat sum.txt | echo $[ $(tr '\n' '+' ) 0 ]
15
```

How does this hack work?

Here, the `tr` command is used to replace `'\n'` with the `'+'` character, hence we form the string `"1+2+3+..5+"`, but at the end of the string we have an extra `+` operator. In order to nullify the effect of the `+` operator, `0` is appended.

`$[ operation ]` performs a numeric operation. Hence, it forms the string as follows:

```
echo $[ 1+2+3+4+5+0 ]
```

If we use a loop to perform the addition by reading numbers from a file, it would take a few lines of code. Here a one-liner does the trick.

`tr` can also be used in this way to get rid of extra newlines as follows:

```
$ cat multi_blanks.txt | tr -s '\n'
line 1
line2
line3
line4
```

In the preceding usage of `tr`, it removes the extra `'\n'` characters into a single `'\n'` (newline character).

Character classes

`tr` can use different character classes as sets. The different classes are as follows:

- `alnum`: Alphanumeric characters
- `alpha`: Alphabetic characters
- `cntrl`: Control (nonprinting) characters
- `digit`: Numeric characters
- `graph`: Graphic characters
- `lower`: Lowercase alphabetic characters
- `print`: Printable characters
- `punct`: Punctuation characters
- `space`: Whitespace characters
- `upper`: Uppercase characters
- `xdigit`: Hexadecimal characters

We can select the required classes and use them as follows:

```
tr [:class:] [:class:]
```

For example:

```
tr '[:lower:]' '[:upper:]'
```

Checksum and verification

Checksum programs are used to generate checksum key strings from the files and verify the integrity of the files later by using that checksum string. A file might be distributed over the network or any storage media to different destinations. Due to many reasons, there are chances of the file being corrupted due to a few bits missing during the data transfer by different reasons. These errors happen most often while downloading the files from the Internet, transferring through a network, CD-ROM damage, and so on.

Hence, we need to know whether the received file is the correct one or not by applying some kind of test. The special key string that is used for this file integrity test is known as a **checksum**.