1 Escaping and Quoting

Flex uses the following metacharacters:

```
. * + ? | / ( ) [ ] { } < > " ^ $ \
```

Additionally whitespace characters are metacharacters. By metacharacter, we mean that in a regular expression, these characters do not refer to themselves. The simplest way to make a metacharacter into a normal character is to escape it with a backslash. For example \* matches a star (and not a sequence of backslashes). This method is the only process to make backslash a regular character. In all other situations (even if quoted or in [...], constructions), the backslash character is a metacharacter.

Inside [...] constructions, ^ is a metacharacter if first and - and ] are metacharacters if not first. Newline, backslash and [ are metacharacters throughout.

Inside "..." constructions, ", \ and newline are the only metacharacters.

2 Bugs!

Find the bugs in the following regular expressions to match C string constants or Pascal comments (hopefully it’s clear which is intended!):

1. "[^\"]*"
2. " ( \[\[\]* \*\* | \[\]\[^\]\]* ) "
3. "[^n] ( \[^\"] | \[^n\]\) * ["]
4. " ( \[^n\] | \.) * "
5. "(* (\[^*\] (\[^*\]) | [^*])|"*")"
6. " (\[^*\]|\[^*\])|"*")"
7. "(* (\[^*\]|"*"[^\"] | [^*]) | [^*])|"*")"

3 Rule of thumb

If you trying to express a negation ("this is not allowed"), make your regular expressions deterministic: make sure that any time there is a choice (especially for |) that the choices never overlap.