

```
-- Week 1, Activity 11:
-- Write a pattern that matches a non-empty list of numerals intercalated with
-- the plus operator (+). A plus operator can only appear between two numerals.
-- Make sure your pattern allows spaces around any element (numerals and
-- operators).
```

```
local lpeg = require "lpeg"
```

```
-- Pattern for 0 or more spaces
spc = lpeg.S(" \n\t")^0
```

```
-- Pattern for 1 or more numbers
num = lpeg.R("09")^1 * spc
```

```
-- Pattern for + operator
plus = lpeg.P("+") * spc
```

```
-- Quick-and-dirty pattern for a non-empty list of numerals intercalated with
-- the plus operator:
p = spc * num * plus * num
```

```
-- Extending the idea to matches binary operators (at most one operator
-- must be present):
mathop = lpeg.S("+-*/%")^-1 * spc
```

```
-- Quick-and-dirty pattern for binary arithmetic:
p2 = spc * num * mathop * num
```

```
-- Pattern test:
-- These must be OK:
print(p:match("1+1"))
print(p:match("    1+1"))
print(p:match("  1+  1"))
print(p:match("  1    +1"))
print(p:match(" 1 + 1"))
print(p:match(" 1 + 1    "))
-- These extended math must be OK:
```

```
print(p2:match("1-1"))
print(p2:match("    1*1"))
print(p2:match("  1/  1"))
print(p2:match("  1    %1"))
print(p2:match(" 1 + 1"))
print(p2:match(" 1 - 1    "))
```

```
-- These must FAIL:
print(p:match("1 1"))
print(p:match("+ 1"))
print(p:match("+"))
print(p:match("123"))
```