```lua
-- Week 1, Activity 9: Basic Constructions
-- How to build simple patterns with LPeg

local lpeg = require "lpeg"

-- lpeg.P: matches a literal string exactly.
print(lpeg.P("hello"):match("hello"))
print(lpeg.P("hella()[]"):match("hella()[]"))

-- lpeg.S: matches any character inside a set.
--         Note that LPeg don't search for the pattern,
--         it only matches in the beginning, a kind of
--         anchored match: it only matches in the beginning
--         of the subject. It's possible to build a search
--         procedure in the pattern itself to find the
--         pattern anywhere in the subject.
print(lpeg.S("aeiou"):match("alo"))     --> 2
print(lpeg.S("aeiou"):match("ei"))      --> 2
print(lpeg.S("aeiou"):match("hello"))   --> nil

-- lpeg.R: matches a range (it's possible to use more
--         than one range).
print(lpeg.R("AZ"):match("Ap"))         --> 2
print(lpeg.R("AZ"):match("("))          --> nil
print(lpeg.R("AZ", "az"):match("aB"))   --> 2

-- lpeg.P(N): matcher N number of character, whatever they are.
--            If the subject does not have that number of
--            characters, the matching fails.
print(lpeg.P(3):match("ABCDE"))         --> 4
print(lpeg.P(3):match(12345))           --> 4
print(lpeg.P(3):match("oi"))            --> nil
```