

Exercise 4.36.

Exercise 3.69 discussed how to generate the stream of *all* Pythagorean triples, with no upper bound on the size of the integers to be searched. Explain why simply replacing `an-integer-between` by `an-integer-starting-from` in the procedure in exercise 4.35 is not an adequate way to generate arbitrary Pythagorean triples. Write a procedure that actually will accomplish this. (That is, write a procedure for which repeatedly typing `try-again` would in principle eventually generate all Pythagorean triples.)

Answer.

By simply replacing `an-integer-between` with `an-integer-starting-from`, the procedure in exercise 4.35 becomes:

```
(define (pythagorean-triples-starting-from n)
  (let ((i (an-integer-starting-from n)))
    (let ((j (an-integer-starting-from i)))
      (let ((k (an-integer-starting-from j)))
        (require (= (+ (* i i) (* j j)) (* k k)))
        (list i j k))))))
```

It is far from adequate for triples of boundless integers, for the `amb` evaluator implements a systematic search in a depth-first way. In other words, it tries all the possible values of k before changing the value of i and j . In particular, if we try to generate all Pythagorean triples using


```
(pythagorean-triples-starting-from 1)
```

our `amb` evaluator will first try to traverse all triples with both i and j equal to 1, and hence will never produce triples with any other value of the first integers.

To handle triples of boundless integers, we need to devise an order of combination that ensure that every element will eventually be reached if we let our program run long enough. An elegant way to accomplish this is with the following `pythagorean-triples-starting-from` procedure:

```
(define (pythagorean-triples-starting-from n)
  (let ((k (an-integer-starting-from n)))
    (let ((i (an-integer-between n k)))
      (let ((j (an-integer-between i k)))
        (require (= (+ (i i) (* j j)) (* k k)))
        (list i j k))))))
```

Since the new `pythagorean-triples-starting-from` ambiguously selects an integer starting from n for k and investigates all the possible value of i and j in between, every element of sequences i and j will find their way into the triple to be investigated, even if all the three sequence is infinite.

*. Creative Commons  2014, Lawrence X. Amlord (颜世敏, aka 颜序).
Email address: informlarry@gmail.com