

Answer. It is obvious that the absolute tolerance of 0.001 is extremely rough when it comes to compute the square root of a relatively small number. Let's take 0.00001 for example:

Instead of putting out 0.0031622776601683794, which is legitimate. The original compound procedure `sqrt` gives a wrong answer with a tolerance over 900%.

```
(sqrt 88888888888888888888888888888888)
>
```

As is mentioned in the problem, in order to get an alternative strategy for implementing **good-enough?** with our limited precision, we should be able to represent the change with a limited proportion of **guess** itself. Well, let's go:

Having the new **good-enough?** conditional defined above, we can easily rewrite the original square-root procedure as the following:

Now let's do some test on the new `sqrt` procedure as before:

By comparing the performance of the new `sqrt` procedure with the original version from the book. We are now sure that the new version of `sqrt` procedure works better for both small and large number.

