CP.1.2.3, Sauer3

Calculate the square roots of the following numbers to eight correct decimal places by using Fixed-Point Iteration as in Example 1.6: (a) 3 (b) 5. State your initial guess and the number of steps needed.

Colab link: https://colab.research.google.com/drive/1Ze9NgNqovpc89Xr3Azta-y-H_I4BArSM

CP.1.2.3, Sauer3, solution, Langou

```
from math import sqrt
```

```
# (a) 3

A = 3.

r = sqrt( A )

x = 2.
true_fwd_rel_error = abs( r - x ) / abs( r )
print( f"{0:2d}", f"{x:.15f}", f"{true_fwd_rel_error:.2e}" )

for i in range(0,4):
  x = ( x + A / x ) / 2.
  true_fwd_rel_error = abs( r - x ) / abs( r )
  print( f"{i+1:2d}", f"{x:.15f}", f"{true_fwd_rel_error:.2e}" )

# I chose x = 2. as the initial guess
# 4 iterations are needed to obtain an answer correct to 8 decimal places
```

```
 0 2.000000000000000 1.55e-01
 1 1.750000000000000 1.04e-02
 2 1.732142857142857 5.31e-05
 3 1.732050810014727 1.41e-09
 4 1.732050807568877 0.00e+00
```

```
# (b) 5

A = 5.

r = sqrt( A )

x = 2.
true_fwd_rel_error = abs( r - x ) / abs( r )
print( f"{0:2d}", f"{x:.15f}", f"{true_fwd_rel_error:.2e}" )
```

```
for i in range(0,4):
  x = ( x + A / x ) / 2.
  true_fwd_rel_error = abs( r - x ) / abs( r )
  print( f"{i+1:2d}", f"{x:.15f}", f"{true_fwd_rel_error:.2e}" )

# I chose x = 2. as the initial guess
# 3 iterations are needed to obtain an answer correct to 8 decimal places
```

```
 0 2.000000000000000 1.06e-01
 1 2.250000000000000 6.23e-03
 2 2.236111111111111 1.93e-05
 3 2.236067977915804 1.86e-10
 4 2.236067977499790 0.00e+00
```