國立臺灣大學

Problem Set — For

Computer Programming (EE3031), F21

Jiun-Lang Huang

EE Department, National Taiwan University

Problem F01: Square Root

One way of computing square roots is the Newton's method. Suppose that you want to compute the square root of x. Let a_i be your i-th guess. a_{i+1}, a better guess, can be obtained by $a_{i+1} = \dfrac{a_i + \dfrac{x}{a_i}}{2}$.

For example, let $x = 5$ and the initial guess of $\sqrt{x}$ be $\dfrac{x}{2}$, i.e., 2.5. We have $a_1 = \dfrac{a_0 + \dfrac{x}{a_0}}{2} = \dfrac{2.5 + \dfrac{5}{2.5}}{2} = 2.25$. By repeating the process, we can obtain a good approximation of $\sqrt{5}$.

- Write a program `square_root.py` that asks the user to enter a positive integer, x. Then, apply the Newton's method (with $a_0 = \frac{x}{2}$) and shows the the first ten guesses.

- Example IO:

```
fin2017 > python3 square_root.py
enter a positive integer: 5
guess 0: 2.50000000
guess 1: 2.25000000
guess 2: 2.23611111
guess 3: 2.23606798
guess 4: 2.23606798
guess 5: 2.23606798
guess 6: 2.23606798
guess 7: 2.23606798
guess 8: 2.23606798
guess 9: 2.23606798
fin2017 >
```