

1 Problem 1

The golden ratio is defined as $\phi = \frac{a}{b} = \frac{a+b}{a}$. This ratio can be solved for algebraically:

$$\begin{aligned} a^2 &= ab + b^2 \\ a^2 - ab - b^2 &= 0 \\ a &= \frac{b \pm \sqrt{b^2 + 4b^2}}{2} \\ \phi = \frac{a}{b} &= \frac{1 + \sqrt{5}}{2}. \end{aligned}$$

In the last step, the plus sign was taken in order to keep with the definition that $\phi > 0$.

The Fibonacci sequence is given by

$$f(n) = \begin{cases} 1 & n = 0, 1 \\ f(n-1) + f(n-2) & n > 1 \end{cases}$$

Consider the ratio

$$r(n) = \frac{f(n)}{f(n-1)}.$$

Assume that $\lim_{n \rightarrow \infty} r(n) = c$. Claim that $c = \phi$.

Proof. (Informal) As n is arbitrarily large, additionally assume

$$\frac{f(n+1)}{f(n)} = \frac{f(n)}{f(n-1)} = c.$$

Then,

$$\begin{aligned} c &= \frac{f(n+1)}{f(n)} \\ &= \frac{f(n-1) + f(n)}{f(n)} \\ &= \frac{1}{c} + 1 \\ c^2 - c - 1 &= 0 \\ c &= \frac{1 + \sqrt{5}}{2} \\ c &= \phi \end{aligned}$$

□

2 Problem 2

This part was different. For this problem, I modified the source code from Problem 1 to calculate $|\phi - r(n)|$ (modified source code included below and in folder). This difference should approach zero as n gets large, as indicated by the previous proof. The results of this calculation are plotted in Figure ???. As shown, $r(n)$ converges to ϕ exponentially with a decay constant of *VALUE*.

P2/Fibo.java

```
1 // fibo.java
2 // fibonacci java class 2
3 // write out fibonacci sequence and ratio
4 // Athanasios Athanassiadis
5 // Complexity, Project 1, F2013
6 // 10/3/2012
7
8 import java.applet.Applet;
9 public class Fibo extends Applet
10 {
11
12     int i,j,k;
13     int n, nsteps, ypos;
14     double r, q;
15     double g = 1.618;
16
17     public void start()
18     {
19         nsteps = 20;
20
21         i=1;
22         j=1;
23         System.out.println(" 0,1,1 ");
24
25         for (n=1; n<=nsteps; n++)
26         {
27             k = i+j;
28             r = (double) k / (double) j;
29             q = Math.abs(g - r);
30             System.out.println(" " + n + ", " + k + ", " + q );
31             i = j;
32             j = k;
33         }
34     }
35 }
36
```