University of Nevada, Las Vegas Computer Science 302 Spring 2017 August 29, 2018

Addendum to the Lecture

The following code:

for(int i = 1; i < n; i=i*i)</pre>

will never halt if n > 1, since *i* will be set to 1 at every iteration.

The asymptotic complexity of the followng code:

for(int i = 0; i*i < n; i++)</pre>

is $\Theta(\sqrt{n})$ because i * i > n when i exceeds \sqrt{n} .

The asymptotic complexity of the followng code:

for(int i = 2; i < n; i=i*i)</pre>

is $\Theta(\log \log n)$. This is much harder to analyze.

- 1. Define variables
 - (a) $m = \log_2 n$
 - (b) $j = \log_2 i$
- 2. Substitute those values into the original code. Note that $\log(i^2) = 2\log i = 2j$.

for(int j = 1; j < m; j = 2*j)</pre>

That is, for a given iteration of the new code the value of j is equal to the value of $\log_2 i$ for the same iteration of the original code.

We already know that the complexity of that fragment is $\Theta(\log m) = \Theta(\log \log n)$.