Assignment 1

Benjamin Jakubowski

June 1, 2016

1. Computing index of maximum element in array

An algorithm to compute the index of the maximum element in an array is given below:

Algorithm 1 Index of maximum element in array

```
1: function Max_index(A)
2:     max_val = A[1]
3:     max_index = 1
4:     for j in [2, ··· , A.length] do:
5:         candidate = A[j]
6:         if max_val < candidate then:
7:             max_val = candidate
8:             max_index = j
       return max_index
```

The loop invariant is as follows; at the start of the **for** loop of lines 4-8, max_val is greater than or equal to all elements in the subarray $A[1, \cdots, j-1]$. We now use this invariant to demonstrate the correctness of this algorithm.

- **Initialization**: First, we must show the loop invariant holds before the first loop iteration, when $j = 2$. The subarray $A[1, \cdots, j-1] = A[1, \cdots, 2-1]$ is just the single element $A[1]$. Since max_val was initialized to $A[1]$ in line 2, the loop invariant clearly holds.
- **Maintainence**: Next, we must show that each iteration maintains the loop invariant. We have two cases, corresponding to the if-statement in line 4:
  - **max_val < candidate**: Then candidate must be greater than or equal to all the elements in the subarray $A[1, \cdots, j-1]$. Thus, setting max_val to candidate in line 7 ensures max_val is greater than or equal to all elements in the subarray $A[1, \cdots, j]$, maintaining the loop invariant before the start of the next iteration.

– **max_val $\geq$ candidate**: Then the if-block (lines 7-8) is not executed, and before the start of the next iteration max_val is greater than or equal to all elements in the subarray $A[1, \cdots, j]$ (since it was assumed to be greater than or equal to all elements in $A[1, \cdots, j-1]$, as well as $A[j]$).

- **Termination**: Finally, the loop terminates when $j = n+1 > A.length = n$. Note at the start of this final iteration, max_val is greater than or equal to all elements in $A[1, \cdots, j-1] = A[1, \cdots, n+1-1] = A[1, \cdots, n]$. Hence we conclude the algorithm finds the maximum value in the array A, and returns its corresponding index.

2. Lucas numbers

The Lucas numbers as defined as follows:

$$L_n = \begin{cases} 2 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ LL_{n-1} + L + n - 2 & \text{if } n > 1 \end{cases}$$

We prove by induction the closed-form expression for the n-th Lucas number:

$$L_n = \varphi^n + (1-\varphi)^n$$

where φ is the golden ratio:

$$\varphi = \frac{1+\sqrt{5}}{2}$$

First, we prove two base cases:
Base case 1: Consider n=2. Then

$$\begin{aligned} L_{n=2} &= L_{n-1} + L_{n-2} \\ &= L_{2-1} + L_{2-2} \\ &= L_1 + L_0 \\ &= 1 + 2 = 3 \end{aligned}$$

Also,

$$\begin{aligned} \varphi^2 + (1-\varphi)^2 &= \varphi^2 + 1 - 2\varphi + \varphi^2 \\ &= 2\varphi^2 - 2\varphi + 1 \\ &= \frac{(1+\sqrt{5})^2}{2} - (1+\sqrt{5}) + 1 \\ &= \frac{1+2\sqrt{5}+5}{2} - \sqrt{5} \\ &= \frac{6}{2} = 3 \end{aligned}$$

Base case 2: Consider n=3. Then

$$\begin{aligned} L_{n=3} &= L_{3-1} + L_{3-2} \\ &= L_2 + L_1 \\ &= 3 + 1 = 4 \end{aligned}$$

Also,

$$\begin{aligned} \varphi^3 + (1-\varphi)^3 &= \varphi^3 + 1 - 3\varphi + 3\varphi^2 - \varphi^3 \\ &= 1 - 3\varphi + 3\varphi^2 \\ &= 1 - 3 \cdot \frac{1+\sqrt{5}}{2} + 3\left(\frac{1+\sqrt{5}}{2}\right)^2 \\ &= 1 - \frac{3}{2} - \frac{3\sqrt{5}}{2} + \frac{3}{4} + \frac{3\sqrt{5}}{2} + \frac{3 \cdot 5}{4} \\ &= -\frac{1}{2} + \frac{18}{4} = 4 \end{aligned}$$

Next, we proceed to the induction step. Assume

$$L_n = \varphi^n + (1-\varphi)^n$$
$$L_{n+1} = \varphi^{n+1} + (1-\varphi)^{n+1}$$

We need to show

$$L_{n+2} = \varphi^{n+2} + (1-\varphi)^{n+2}$$

First, note

$$\begin{aligned} \varphi^2 &= \left(\frac{1+\sqrt{5}}{2}\right)^2 \\ &= \frac{1+2\sqrt{5}+5}{4} \\ &= \frac{6+2\sqrt{5}+5}{4} \\ &= \frac{3+\sqrt{5}}{2} \\ &= 1 + \frac{1+\sqrt{5}}{2} \\ &= 1 + \varphi \end{aligned}$$

Then

$$\begin{aligned} \varphi^{k+2} &= \varphi^k \cdot \varphi^2 \\ &= \varphi^k \cdot (1+\varphi) \\ &= \varphi^k + \varphi^{k+1} \end{aligned}$$

Also,

$$\begin{aligned}
(1-\varphi)^2 &= \left(1 - \frac{1+\sqrt{5}}{2}\right)^2 \\
&= 1 - 2\cdot\frac{1+\sqrt{5}}{2} + \left(\frac{1+\sqrt{5}}{2}\right)^2 \\
&= 1 - 1 - \sqrt{5} + \frac{1+2\sqrt{5}+5}{4} \\
&= -\sqrt{5} + \frac{3}{2} + \frac{\sqrt{5}}{2} \\
&= \frac{3-\sqrt{5}}{2} \\
&= 2 - \frac{1-\sqrt{5}}{2} \\
&= 1 + (1-\varphi)
\end{aligned}$$

So

$$\begin{aligned}
(1-\varphi)^{k+2} &= (1-\varphi)^k(1-\varphi)^2 \\
&= (1-\varphi)^k\left(1+(1-\varphi)\right) \\
&= (1-\varphi)^k + (1-\varphi)^{k+1}
\end{aligned}$$

Now we put these two expressions together to complete the proof:

$$\begin{aligned}
L_{n+2} &= L_{n+1} + L_n \\
&= \varphi^{n+1} + (1-\varphi)^{n+1} + \varphi^n + (1-\varphi)^n \\
&= \varphi^{n+1} + \varphi^n + (1-\varphi)^{n+1} + (1-\varphi)^n \\
&= \varphi^{n+2} + (1-\varphi)^{n+2}
\end{aligned}$$

completing the proof.

3. Properties of Θ

a. Being in Θ is an equivalence relation

To prove that being in Θ is an equivalence relation, we must show it is reflexive, symmetric, and transitive. We first show reflexivity.
Consider a function f, and assume it is asymptotically non-negative. Then let $c_1 = 1/2$, $c_2 = 3/2$, and n_0 be some constant such that $f(n) \geq 0$ for all $n > n_0$ (by asymptotic non-negativity). Then

$$0 \leq c_1 f(n) \leq f(n) \leq 3/2 f(n) \qquad \forall n \geq n_0$$

Thus, $f(n) \in \Theta(f(n))$.
Now we show symmetry- consider two functions f and g. First, assume $f \in \Theta(g)$. Then there exists $c_1, c_2, n_0 > 0$ such that

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \qquad \forall n > n_0$$

Then note

$$0 \leq g(n) \leq 1/c_1 f(n) \qquad \forall n > n_0$$
$$0 \leq 1/c_2 f(n) \leq g(n) \qquad \forall n > n_0$$

So, putting these inequalities together yeilds

$$0 \leq 1/c_2 f(n) \leq g(n) \leq 1/c_1 f(n) \qquad \forall n > n_0$$

Thus $g \in \Theta(f(n))$.
Now note the proof is essentially equivalent for the other direction (namely, assuming $g \in \Theta(f)$, that $g \in \Theta(f)$); all we'd do is switch the function names and use new constants. Thus this proof is omitted.
Finally, we show transitivity.
Assume $f \in \Theta(g)$ and $g \in \Theta(h)$. We show $f \in \Theta(h)$. Since $f \in \Theta(g)$, there exists $c_1, c_2, n_0 > 0$ such that

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \qquad \forall n > n_0$$

Since $g \in \Theta(h)$, there exists $c_3, c_4, n_1 > 0$ such that

$$0 \leq c_3 h(n) \leq g(n) \leq c_4 h(n) \qquad \forall n > n_1$$

But then substituting yields

$$0 \leq c_1 \cdot c_3 h(n) \leq f(n) \leq c_2 \cdot c_4 h(n) \qquad \forall n \geq \max\{n_0, n_1\}$$

Hence $f \in \Theta(h)$.
Thus, having shown reflexivity, symmetry, and transitivity, we've shown Θ is an equivalence relation.

b. Maximum of two functions is in Θ of their sum

Consider f_1 and f_2 (again assuming asymptotic non-negativity). We aim to show

$$\max\{f_1, f_2\} \in \Theta(f_1 + f_2)$$

First, let $f = \max\{f_1, f_2\}$ to easy notation. Then note

$$0 \leq f(n) \leq f_1(n) + f_2(n) \qquad \forall n > n_0$$

(again using asymptotic non-negativity to chose n_0 such that for all $n > n_0, f_1(n) \geq 0$ and $f_2(n) \geq 0$).

Moreover, note

$$0 \leq 1/2\,(f_1(n) + f_2(n)) \leq f(n) \qquad \forall n > n_0$$

Since either $f(n) = f_1(n) > f_2(n)$ or $f(n) = f_2(n) > f_1(n)$. But then,

$$0 \leq 1/2\,(f_1(n) + f_2(n)) \leq f(n) \leq f_1(n) + f_2(n) \qquad \forall n > n_0$$

Hence $f(n) = \max\{f_1(n), f_2(n)\} \in \Theta\,(f_1(n) + f_2(n))$.

c. Sum of two functions is in Θ of their maximum

Now we show the sum of two functions is in Θ of their maximum in two lines!

1. Being in Θ is an equivalence relation, so it's symmetric.
2. Thus $\max\{f_1(n), f_2(n)\} \in \Theta\,(f_1(n) + f_2(n)) \implies (f_1(n) + f_2(n)) \in \Theta(\max\{f_1(n), f_2(n)\})$.

4. Ranking function forms by order of growth

The given forms are ranked in order of growth below (from slowest to fastest):

1. Constant
2. Logarithmic
3. Linear
4. Linearithmic
5. Polynomial
6. Exponential

To prove this ranking, we proceed pairwise. Throughout, recall $g(n) \in o(f(n))$ if

$$\lim_{n\to\infty} \frac{g(n)}{f(n)} = 0$$

Additionally, note we are treating logs as $\log_e$, since a change of basis only introduces a constant term. Additionally, we are ignoring all constants (since they don't change the limiting behavior).

Constant vs. Logarithmic

Let $g(n) = c$ and $f(n) = \log(n)$. Then

$$\lim_{n\to\infty} \frac{g(n)}{f(n)} = \lim_{n\to\infty} \frac{c}{\log(n)} = 0$$

Logarthimic vs. Linear

Let $g(n) = \log(n)$ and $f(n) = n$. Then

$$\begin{aligned}
\lim_{n\to\infty} \frac{g(n)}{f(n)} &= \lim_{n\to\infty} \frac{\log(n)}{n} \\
&= \lim_{n\to\infty} \frac{(\log(n))'}{(n)'} && \text{by l'Hopital's rule} \\
&= \lim_{n\to\infty} \frac{1}{n} = 0
\end{aligned}$$

Linear vs. Linearithmic

Let $g(n) = n$ and $f(n) = n\log(n)$. Then

$$\begin{aligned}
\lim_{n\to\infty} \frac{g(n)}{f(n)} &= \lim_{n\to\infty} \frac{n}{n\log(n)} \\
&= \lim_{n\to\infty} \frac{1}{\log(n)} = 0
\end{aligned}$$

Linearithmic vs. Polynomial

Let $g(n) = n\log(n)$ and $f(n) = n^d$ for some $d > 1$. Then

$$\begin{aligned}
\lim_{n\to\infty} \frac{g(n)}{f(n)} &= \lim_{n\to\infty} \frac{n\log(n)}{n^d} \\
&= \lim_{n\to\infty} \frac{\log(n)}{n^{d-1}} \\
&= \lim_{n\to\infty} \frac{(\log(n))'}{(n^{d-1})'} \\
&= \lim_{n\to\infty} \frac{n^{-1}}{(d-1)n^{d-2}} \\
&= \lim_{n\to\infty} \frac{1}{d-1} n^{-1-(d-2)} \\
&= \lim_{n\to\infty} \frac{1}{d-1} n^{1-d} = 0 && \text{since d>1}
\end{aligned}$$

Polynomial vs. Exponential

Let $g(n) = n^d$ for some $d > 1$ and $f(n) = a^n$ for some $a > 1$. Then

$$\begin{aligned}
\lim_{n\to\infty} \frac{g(n)}{f(n)} &= \lim_{n\to\infty} \frac{n^d}{a^n} \\
&= \lim_{n\to\infty} \frac{(n^d)'}{(a^n)'} \\
&= \lim_{n\to\infty} \frac{d \cdot n^{d-1}}{\log(a)a^n} \\
&\vdots \qquad \lceil d \rceil \text{ more applications of l'Hopital's Rule} \\
&= \lim_{n\to\infty} \frac{k \cdot n^{d-\lceil d \rceil -1}}{\log(a)^{\lceil d \rceil +1} a^n} \qquad \text{where } k \text{ is just a constant} \\
&= 0
\end{aligned}$$