CS350 — Winter 2019
Solutions for Homework 0

Problem 1

Read the descriptions of the three algorithms for computing GCD that appear in Levitin §1.1: Euclid's Algorithm, the Consecutive Integer checking algorithm, and the middle-school procedure.

Implement Euclid's algorithm and the consecutive integer checking algorithm in your favorite programming language. Measure their execution times, and compare how long each takes to run on pairs of natural numbers of 8, 16, 32, 64, 128, 256 and 512 digits. (Depending on your language, you may need to use a special package to represent large integers.)

Solution. (2pt for Euclid's and 2pt for Consecutive Integer checking algorithm)

What I was expecting for this question, in addition to a table of timings, was a short explanation of the environment, including the language, bignum package (if necessary) and timing regime that you used to get your results. For the consecutive integer algorithm, it's infeasible to measure the time taken even for 16-digit numbers, let alone larger ones. Euclid's Algorithm, in contrast, is feasible even for 256-digit numbers.

Problem 2

Use induction to prove that

$$2^{n+1} - 1 = 2^0 + 2^1 + 2^2 + \cdots + 2^n$$

(1pt for basecase, 1pt for induction hypothesis)

(1pt for using Induction Hypothesis, 1pt for proving the required equality)

Proof. We will use induction over the natural numbers $\mathbb{N}$, ordered by $<$. As induction hypothesis, we use $H(n)$ defined as

$$H(n) \triangleq 2^{n+1} - 1 = 2^0 + 2^1 + 2^2 + \cdots + 2^n \tag{1}$$

Base case: The base case is $H(0)$:

$$H(0) \equiv \quad 2^1 - 1 = 2^0 \quad \text{[substituting } n = 0 \text{ in (1)]} \tag{2}$$

$$1 = 1 \quad \text{[arithmetic]} \tag{3}$$

Induction step: For all $a > 0$ in $\mathbb{N}$, we must prove $H(a)$ while assuming

$$H(a-1) \tag{4}$$

$$\begin{aligned}
H(a-1) \equiv \quad 2^a - 1 &= 2^0 + 2^1 + 2^2 + \cdots + 2^{a-1} && [(1)] && (5)\\
2(2^a - 1) &= 2(2^0 + 2^1 + 2^2 + \cdots + 2^{a-1}) && [\text{multiply by } 2] && (6)\\
2 \cdot 2^a - 2 &= 2 \cdot 2^0 + 2 \cdot 2^1 + 2 \cdot 2^2 + \cdots + 2 \cdot 2^{a-1}) && [\text{expand}] && (7)\\
2^{a+1} - 2 &= 2^1 + 2^2 + \cdots + 2^a && [\text{def of power}] && (8)\\
2^{a+1} - 1 &= 1 + 2^1 + 2^2 + \cdots + 2^a && [\text{add } 1] && (9)\\
2^{a+1} - 1 &= 2^0 + 2^1 + 2^2 + \cdots + 2^a && [2^0 = 1] && (10)\\
&\Rightarrow H(a) && [(1)] && (11)
\end{aligned}$$

□

Problem 3

Prove that

$$a^{\log_b x} = x^{\log_b a}$$

(1pt stating your assumptions, 1pt for proof of work)

(1pt for proving the eq, 1pt for stating the properties used to prove the eq)

Proof. Since this is an elementary property of logarithms, it's not appropriate to use more complex properties, like the base-change rule, in the proof. This theorem is just a consequence of commutativity of multiplication and the property that $\log x^n = n \log x$.

$$\begin{aligned}
a^{\log_b x} &= a^{\log_b x} && [\text{identity}] && (12)\\
\log_b a^{\log_b x} &= \log_b \left(a^{\log_b x}\right) && [\log_b \text{ of } (12)] && (13)\\
&= (\log_b x)(\log_b a) && [\text{property of log}] && (14)\\
&= (\log_b a)(\log_b x) && [\text{commutativity of } \times] && (15)\\
&= \log_b x^{\log_b a} && [\text{property of log}] && (16)\\
a^{\log_b x} &= x^{\log_b a} && [\text{raise } b \text{ to } 16] && (17)
\end{aligned}$$

□

Problem 4

Prove that

$$\log_x ab = \log_x a + \log_x b$$

(1pt stating your assumptions, 1pt for proof of work)

(1pt for proving the eq, 1pt for stating the properties used to prove the eq)

Proof. Let $m \triangleq \log_x a$ and $n \triangleq \log_x b$. Then:

$$\begin{aligned}
x^m &= a && \text{[raise } x \text{ to def of } m\text{]} && (18)\\
x^n &= b && \text{[raise } x \text{ to def of } n\text{]} && (19)\\
x^m * x^n &= a * b && \text{[Multiplication]} && (20)\\
x^{m+n} &= ab && \text{[property of exponents]} && (21)\\
\log_x x^{m+n} &= \log_x ab && \text{[log applied to both sides]} && (22)\\
m + n &= \log_x ab && \text{[def of log]} && (23)\\
\log_x a + \log_x b &= \log_x ab && \text{[subst. def of } m \text{ and } n\text{]} && (24)
\end{aligned}$$

□

Problem 5

Use induction to prove that

$$2^n < n! \text{ is true } \forall\ \mathbb{N} \text{ where } n \geqslant 4$$

(1pt for basecase, 1pt for induction hypothesis)

(1pt for using I.H, 1pt for proving the equation)

Proof. We'll use induction over the natural numbers $\mathbb{N}$, which are $\geqslant 4$. As induction hypothesis, we use $H(n)$ defined as

$$H(n) \triangleq 2^n < n! \tag{25}$$

Base case: The base case is $H(4)$:

$$\begin{aligned}
H(4) \equiv \quad & 2^4 < 4! && \text{[substituting } n = 4 \text{ in (25)]} && (26)\\
& 16 < 24 && \text{[inequalities]} && (27)
\end{aligned}$$

Induction step: For all $a > 4$ in $\mathbb{N}$, we must prove $H(a)$ while assuming

$$H(a-1) \tag{28}$$

$$
\begin{aligned}
H(a-1) \equiv \quad & 2^{a-1} < (a-1)! && [28] && (29)\\
& 2(2^{a-1}) < 2(a-1)! && [\text{multiply by } 2] && (30)\\
& 2^{a} < 2(a-1)! && [\text{def of power}] && (31)\\
& 2(a-1)! < a(a-1)! && [\text{by } 28,\ a > 4] && (32)\\
& 2^{a} < a! && [\text{def of factorial}] && (33)\\
& \Rightarrow H(a) && [28] && (34)
\end{aligned}
$$

$\square$