Applied Formal Methods: Homework #0

Due on Aug. 24 at Midnight

Professor Rozier : T R 11:00-12:15

Liwei Tsai

Problem 1

Give an appropriate positive constant c such that $f(n) \leq c \cdot g(n)$ for all $n > 1$.

1. $f(n) = n^2 + n + 1$, $g(n) = 2n^3$
2. $f(n) = n\sqrt{n} + n^2$, $g(n) = n^2$
3. $f(n) = n^2 - n + 1$, $g(n) = n^2/2$

Solution
We solve each solution algebraically to determine a possible constant c.

Part One

$$\begin{aligned} n^2 + n + 1 &= \\ &\leq n^2 + n^2 + n^2 \\ &= 3n^2 \\ &\leq c \cdot 2n^3 \end{aligned}$$

Thus a valid c could be when $c = 2$.

Part Two

$$\begin{aligned} n^2 + n\sqrt{n} &= \\ &= n^2 + n^{3/2} \\ &\leq n^2 + n^{4/2} \\ &= n^2 + n^2 \\ &= 2n^2 \\ &\leq c \cdot n^2 \end{aligned}$$

Thus a valid c is $c = 2$.

Part Three

$$\begin{aligned} n^2 - n + 1 &= \\ &\leq n^2 \\ &\leq c \cdot n^2/2 \end{aligned}$$

Thus a valid c is $c = 2$.

Problem 2

Let $\Sigma = \{0, 1\}$. Construct a DFA A that recognizes the language that consists of all binary numbers that can be divided by 5.

Let the state q_k indicate the remainder of k divided by 5. For example, the remainder of 2 would correlate to state q_2 because $7 \mod 5 = 2$.

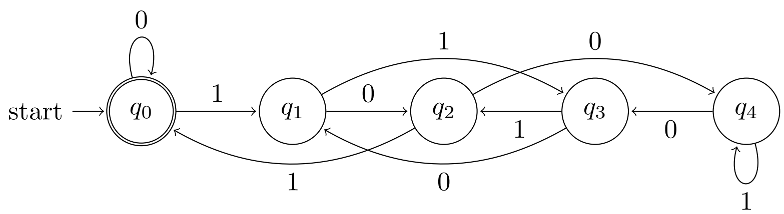

Figure 1: DFA, A, this is really beautiful, ya know?

Justification

Take a given binary number, x. Since there are only two inputs to our state machine, x can either become $x0$ or $x1$. When a 0 comes into the state machine, it is the same as taking the binary number and multiplying it by two. When a 1 comes into the machine, it is the same as multipying by two and adding one.

Using this knowledge, we can construct a transition table that tell us where to go:

	$x \mod 5 = 0$	$x \mod 5 = 1$	$x \mod 5 = 2$	$x \mod 5 = 3$	$x \mod 5 = 4$
$x0$	0	2	4	1	3
$x1$	1	3	0	2	4

Therefore on state q_0 or ($x \mod 5 = 0$), a transition line should go to state q_0 for the input 0 and a line should go to state q_1 for input 1. Continuing this gives us the Figure 1.

Problem 3

Write part of **Quick-Sort**($list, start, end$)

function QUICK-SORT($list, start, end$)
  if $start \geq end$ **then**
    return
  end if
  $mid \leftarrow$ PARTITION($list, start, end$)
  QUICK-SORT($list, start, mid - 1$)
  QUICK-SORT($list, mid + 1, end$)
end function

Algorithm 1: Start of QuickSort

Problem 4

Suppose we would like to fit a straight line through the origin, i.e., $Y_i = \beta_1 x_i + e_i$ with $i = 1, \ldots, n$, $\mathrm{E}[e_i] = 0$, and $\mathrm{Var}[e_i] = \sigma_e^2$ and $\mathrm{Cov}[e_i, e_j] = 0, \forall i \neq j$.

Part A

Find the least squares esimator for $\hat{\beta}_1$ for the slope β_1.

Solution

To find the least squares estimator, we should minimize our Residual Sum of Squares, RSS:

$$\begin{aligned} RSS &= \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2 \\ &= \sum_{i=1}^{n} \left(Y_i - \hat{\beta}_1 x_i\right)^2 \end{aligned}$$

By taking the partial derivative in respect to $\hat{\beta}_1$, we get:

$$\frac{\partial}{\partial \hat{\beta}_1}(RSS) = -2 \sum_{i=1}^{n} x_i (Y_i - \hat{\beta}_1 x_i) = 0$$

This gives us:

$$\begin{aligned} \sum_{i=1}^{n} x_i (Y_i - \hat{\beta}_1 x_i) &= \sum_{i=1}^{n} x_i Y_i - \sum_{i=1}^{n} \hat{\beta}_1 x_i^2 \\ &= \sum_{i=1}^{n} x_i Y_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 \end{aligned}$$

Solving for $\hat{\beta}_1$ gives the final estimator for β_1:

$$\hat{\beta}_1 = \frac{\sum x_i Y_i}{\sum x_i^2}$$

Problem 4 continued on next page. . .

Part B

Calculate the bias and the variance for the estimated slope $\hat{\beta}_1$.

Solution

For the bias, we need to calculate the expected value $\mathrm{E}[\hat{\beta}_1]$:

$$
\begin{aligned}
\mathrm{E}[\hat{\beta}_1] &= \mathrm{E}\left[\frac{\sum x_i Y_i}{\sum x_i^2}\right] \\
&= \frac{\sum x_i \mathrm{E}[Y_i]}{\sum x_i^2} \\
&= \frac{\sum x_i (\beta_1 x_i)}{\sum x_i^2} \\
&= \frac{\sum x_i^2 \beta_1}{\sum x_i^2} \\
&= \beta_1 \frac{\sum x_i^2 \beta_1}{\sum x_i^2} \\
&= \beta_1
\end{aligned}
$$

Thus since our estimator's expected value is β_1, we can conclude that the bias of our estimator is 0.

For the variance:

$$
\begin{aligned}
\mathrm{Var}[\hat{\beta}_1] &= \mathrm{Var}\left[\frac{\sum x_i Y_i}{\sum x_i^2}\right] \\
&= \frac{\sum x_i^2}{\sum x_i^2 \sum x_i^2} \mathrm{Var}[Y_i] \\
&= \frac{\sum x_i^2}{\sum x_i^2 \sum x_i^2} \mathrm{Var}[Y_i] \\
&= \frac{1}{\sum x_i^2} \mathrm{Var}[Y_i] \\
&= \frac{1}{\sum x_i^2} \sigma^2 \\
&= \frac{\sigma^2}{\sum x_i^2}
\end{aligned}
$$

Problem 5

Prove a polynomial of degree k, $a_k n^k + a_{k-1} n^{k-1} + \ldots + a_1 n^1 + a_0 n^0$ is a member of $\Theta(n^k)$ where $a_k \ldots a_0$ are nonnegative constants.

Proof. To prove that $a_k n^k + a_{k-1} n^{k-1} + \ldots + a_1 n^1 + a_0 n^0$, we must show the following:

$$\exists c_1 \exists c_2 \forall n \geq n_0,\ c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$$

For the first inequality, it is easy to see that it holds because no matter what the constants are, $n^k \leq a_k n^k + a_{k-1} n^{k-1} + \ldots + a_1 n^1 + a_0 n^0$ even if $c_1 = 1$ and $n_0 = 1$. This is because $n^k \leq c_1 \cdot a_k n^k$ for any nonnegative constant, c_1 and a_k.

Taking the second inequality, we prove it in the following way. By summation, $\sum_{i=0}^{k} a_i$ will give us a new constant, A. By taking this value of A, we can then do the following:

$$\begin{aligned} a_k n^k + a_{k-1} n^{k-1} + \ldots + a_1 n^1 + a_0 n^0 = & \\ & \leq (a_k + a_{k-1} \ldots a_1 + a_0) \cdot n^k \\ & = A \cdot n^k \\ & \leq c_2 \cdot n^k \end{aligned}$$

where $n_0 = 1$ and $c_2 = A$. c_2 is just a constant. Thus the proof is complete. □

Problem 6

What are the connectives in propositional logic?

Connectives

Connective	Pronunciation	Meaning	Alternative pronunciations / notations
$\neg$	not	$\neg a$: a is false	$-a$; $!a$
$\wedge$	and	$a \wedge b$; a and b are both true	$a*b$; ab; $a\&\&b$; $a\&b$
$\vee$	or	$a \vee b$: at least one of $\{a, b\}$ is true	$a+b$; $a\|\|b$; $a\|b$
$\Rightarrow$	implies	$a \Rightarrow b$: equivalent to $\neg a \vee b$	$a \rightarrow b$; $a \supset b$; if a then b; a only if b; b if a; b is necessary for a; a is sufficient for b

Problem 7

This is not a problem. It is an example of a table-formatted proof with subproofs that have subproofs.

1	$H\text{-}has\text{-}2 \Rightarrow P\text{-}unsafe \wedge G\text{-}unsafe \vee J\text{-}unsafe \wedge P\text{-}unsafe \vee G\text{-}unsafe \wedge J\text{-}unsafe$	WaterWorld axiom, choosing a grouping of the ternary $\vee$, as justified by $\vee$ commutativity
2	subproof: $\vdash H\text{-}has\text{-}2$	

Problem 7 continued on next page...

2.a		$\vdash H\text{-}has\text{-}2 \wedge J\text{-}safe$		Premise
2.b		$\vdash H\text{-}has\text{-}2$		$\wedge$Elim (left), line 2.a
3	$P\text{-}unsafe \wedge G\text{-}unsafe \vee J\text{-}unsafe \wedge P\text{-}unsafe \vee G\text{-}unsafe \wedge J\text{-}unsafe$			$\Rightarrow$Elim, lines 1, 3
4	subproof: $\vdash \neg J\text{-}unsafe$			
4.a		$H\text{-}has\text{-}2 \wedge J\text{-}safe$		Premise
4.b		$J\text{-}safe$		$\wedge$Elim (right), line 4.a
4.c		$J\text{-}safe \Rightarrow \neg J\text{-}unsafe$		WaterWorld axiom
4.d		$\neg J\text{-}unsafe$		$\Rightarrow$Elim, lines 4.b, 4.c
5	subproof: $\vdash P\text{-}unsafe \wedge G\text{-}unsafe$			
5.a		subproof: $G\text{-}unsafe \wedge J\text{-}unsafe \vdash$ `false`		
5.a.i			$G\text{-}unsafe \wedge J\text{-}unsafe$	Premise for subproof
5.a.ii			$J\text{-}unsafe$	$\wedge$Elim (right), line 5.a.1
5.a.iii			`false`	`false`Intro, lines 4, 5.a.2
5.b		$\neg(G\text{-}unsafe \wedge J\text{-}unsafe)$		RAA, line 5.a
5.c		$P\text{-}unsafe \wedge G\text{-}unsafe \vee J\text{-}unsafe \wedge P\text{-}unsafe$		CaseElim (right), lines 3, 5.b
5.d		subproof: $J\text{-}unsafe \wedge P\text{-}unsafe \vdash$ `false`		
5.d.i			$J\text{-}unsafe \wedge P\text{-}unsafe$	Premise for subproof
5.d.ii			$J\text{-}unsafe$	$\wedge$Elim (left), line 5.d.1
5.d.iii			`false`	`false`Intro, lines 4, 5.d.2
5.e		$\neg J\text{-}unsafe \wedge P\text{-}unsafe)$		RAA, line 5.d
5.f		$P\text{-}unsafe \wedge G\text{-}unsafe$		CaseElim (right), lines 5.c, 5.e
6	$G\text{-}unsafe$			$\wedge$Elim (right), line 5

Problem 8

Translate the following English requirement into a Linear Temporal Logic (LTL) formula φ: "The value of p must oscillate at every tick of the system clock."

$$\varphi = \Box((p \wedge \mathcal{X}\neg p) \vee (\neg p \wedge \mathcal{X}p))$$

This formula describes a *computation*; let's call it π. Draw the first 14 time steps of computation π, showing the values of p at each time step. Assume p is initialized to 0.

We draw the computation for this formula in Figure 2.

$$\varphi = \Box((p \wedge \mathcal{X}\neg p) \vee (\neg p \wedge \mathcal{X} p))$$

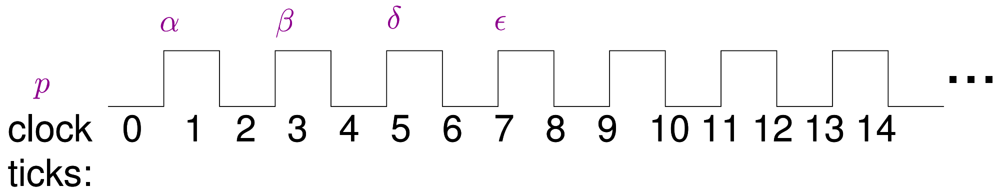

Figure 2: A timing diagram for a computation π with proposition p, where p is initialized to 0. The propositional variable's value during the trace is depicted as a line. When the line is high, the proposition is true; when the line is low, it is false. We can label parts of the diagram with beautiful LaTeXlabels as well that allow us to more easily refer to them in the text: α, β, δ, ϵ, $\ldots$

Problem 18

Evaluate $\sum_{k=1}^{5} k^2$ and $\sum_{k=1}^{5}(k-1)^2$.

Problem 19

Find the derivative of $f(x) = x^4 + 3x^2 - 2$

Problem 6

Evaluate the integrals $\int_0^1 (1 - x^2)\mathrm{d}x$ and $\int_1^\infty \frac{1}{x^2}\mathrm{d}x$.