Problem Set 2

Due: March 3

Reading:

- Chapter 3.6. *Predicate Formulas*,
- Chapter 4. *Mathematical Data Types* through 4.4. *Binary Relations*,
- Chapter 5. *Induction*.

These assigned readings do **not** include the Problem sections. (Many of the problems in the text will appear as class or homework problems.)

Problem 1.
Using predicate formulas built solely from subformulas of the form $a\ R\ b$, express the following assertions.

(a) R is a surjection [>= 1 in].

(b) R is a function [<= 1 out].

Using predicate formulas built solely from subformulas of the form $x \in y$, express the following assertions about sets:

(c) $x = y - z$.

(d) $x = \emptyset$.

Problem 2.
Let A, B and C be sets.

$$A \cup B \cup C = (A - B) \cup (B - C) \cup (C - A) \cup (A \cap B \cap C) \tag{1}$$

by reducing the set-theoretic equality to a valid propositional formula, as in the text.

Problem 3. **(a)** Prove by induction that a $2^n \times 2^n$ courtyard with a 1×1 statue of Bill in *a corner* can be covered with L-shaped tiles. (Do not assume or reprove the (stronger) result of Theorem 5.1.2 that Bill can be placed anywhere. The point of this problem is to show a different induction hypothesis that works.)

(b) Use the result of part (a) to prove the original claim that there is a tiling with Bill in the middle.

Problem 4.
For any binary string α let $\text{num}(\alpha)$ be the nonnegative integer it represents in binary notation. For example, $\text{num}(\texttt{10}) = 2$, and $\text{num}(\texttt{0101}) = 5$.

An $n+1$-*bit adder* adds two $n+1$-bit binary numbers. More precisely, an $n+1$-bit adder takes two length $n+1$ binary strings

$$\alpha_n ::= a_n \dots a_1 a_0,$$
$$\beta_n ::= b_n \dots b_1 b_0,$$

and a binary digit c_0 as inputs, and produces a length-$(n+1)$ binary string

$$\sigma_n ::= s_n \dots s_1 s_0,$$

and a binary digit c_{n+1} as outputs, and satisfies the specification:

$$\text{num}(\alpha_n) + \text{num}(\beta_n) + c_0 = 2^{n+1} c_{n+1} + \text{num}(\sigma_n). \tag{2}$$

There is a straighforward way to implement an $n+1$-bit adder as a digital circuit: an $n+1$-bit *ripple-carry circuit* has $1 + 2(n+1)$ binary inputs

$$a_n, \dots, a_1, a_0, b_n, \dots, b_1, b_0, c_0,$$

and $n+2$ binary outputs,

$$c_{n+1}, s_n, \dots, s_1, s_0.$$

As in Problem 3.6, the ripple-carry circuit is specified by the following formulas:

$$s_i ::= a_i \ \text{XOR} \ b_i \ \text{XOR} \ c_i \tag{3}$$
$$c_{i+1} ::= (a_i \ \text{AND} \ b_i) \ \text{OR} \ (a_i \ \text{AND} \ c_i) \ \text{OR} \ (b_i \ \text{AND} \ c_i),. \tag{4}$$

for $0 \le i \le n$.

(a) Verify that definitions (3) and (4) imply that

$$a_n + b_n + c_n = 2c_{n+1} + s_n. \tag{5}$$

for all $n \in \mathbb{N}$.

(b) Prove by induction on n that an $n+1$-bit ripple-carry circuit really is an $n+1$-bit adder, that is, its outputs satisfy (2).

Hint: You may assume that, by definition of binary representation of integers,

$$\text{num}(\alpha_{n+1}) = a_{n+1} 2^{n+1} + \text{num}(\alpha_n). \tag{6}$$