Thinking Programs: Exercises

Chapter 4: Building Models

Please ensure that your definitions of constants, functions, and numbers are indeed *well-formed* according to the criteria stated in this book.

1. Formally define the following notions (constants, functions, predicates) over the set $\mathbb{N} = \{0, 1, 2, \ldots\}$ of natural numbers:

   - Number m *divides* n if m multiplied with p gives n, for some number p. Numbers q and r are the *quotient and remainder* of m divided by n if $m = n \cdot q + r$ and $r < n$. The *quotient* q of m divided by n is that number q such that q and r are the quotient and remainder of m divided by n for some number r. The *remainder* r of m divided by n is that number r such that such that q and r are the quotient and remainder of m divided by n for some number q.
   - The *predecessor* of n is that number m such that its successor $m + 1$ is n (if such a number exists). The *natural difference* of m and n is that number d that added to n yields m (if such a number exists). The *natural square root* of n is the largest natural number whose square is less than equal n (which always exists).
   - Number p is a *prime* number if it is greater equal 2 and the only divisors of p are 1 and p itself. p is a *prime factor* of n if p is prime and divides n. The *prime factor set* of n is the set of all prime factors of n. The *prime set up to* n is the set of all prime numbers less than equal n. The *next prime number* after n is the smallest prime number greater equal n. Numbers m and n are *relatively prime* if they have no common prime factors. The *prime factorization of* n is that (non-strictly) increasing sequence of prime numbers whose product is n.
   - The *decimal representation of* n is the shortest sequence of decimal digits that in the decimal number system represents n. The *value* of a sequence s of numerical digits in base b is the number denoted by s in the positional number system with that base (lookup the detailed mathematical definitions in the literature).

2. Let S be some finite set. We define a set *Graph* of labeled tuples as follows:

   $$\mathit{Graph} := \{\langle v\colon V, e\colon E\rangle \mid V \subseteq S \wedge E \subseteq V \times V\}$$

   We call every element $G \in \mathit{Graph}$ a (directed) *graph* with *vertices* $G.v$ and *edges* $G.e$. Based on this, formalize the following definitions of some basic notions (constants, functions, predicates) in the theory of directed graphs:

   - Let *empty* be that graph that has no vertices and no edges. The *size* of a graph G is the sum of the number of its vertices and the number of its edges. G is *symmetric* if it contains for every edge also the corresponding inverse edge.

- In graph G, a vertex v_1 is *directly connected* to vertex v_2 if there is in G an edge from v_1 to v_2. Vertices v_1 and v_2 are *adjacent*, if v_1 is directly connected to v_2 or v_2 is directly connected to v_1.
- In graph G, the *successor set* of vertex v is the set of all vertices to which v is directly connected. The *outdegree* of v is the size o the successor set of v. The *predecessor set* of v is the set of all vertices that are directly connected to v. The *indegree* of v is the size of the predecessor set of v. The *neighbor set* of v is the set of all vertices adjacent to v. The *degree* of v is the size of ifs neighbor set (or, equivalently the sum of its indegree and its outdegree). Graph g is *regular* if all its nodes have the same degree.
- A sequence $p \in S^*$ is a *path* in G, if every element in p is a vertex of G and every successive pair of elements in p is connected by an edge of G. p is a *path from v_1 to v_2* if p is a path whose first element is v_1 and whose last element is v_2. In graph G, a vertex v_1 is *connected* to a vertex v_2 if there is some path in G from v_1 to v_2. Graph G is (strongly) *connected* if all pairs of nodes in G are connected.
- In graph G, sequence p is a *cycle* if p is a path in G that has at least two vertices where its first and its last vertex are the same. Graph G is *acyclic* if it does not contain any cycles. Graph G is a *tree* if it is connected and acyclic.

3. Formally define the following notions (constants, functions, predicates) over the theory of (set-theoretic) functions (in these definitions you may omit all type signatures):

   - f is a *partial function* from A to B (written as $f\colon A \to_\perp B$) if f is a subset of $A \times B$ that does not contain two different tuples with the same first component. The *domain* of f is the set with every value a such that f contains a tuple with a as its first component. The *range* of f is the set with every value b such that f contains a tuple with b as the second component. $apply(f, a)$ is that value b such that the tuple with first component a and second component b is in f (if such a tuple exists). f is a *total function* from A to B (written as $f\colon A \to B$) if f is a partial function from A to B whose domain is A.
   - f is *injective* from A to B if it is a total function from A to B such that f does not contain two tuples with the same second argument. f is *surjective* from A to B if it is a total function from A to B such that f contains for every element in B a tuple with that element as second component. f is *bijective* from A to B if it is both *injective* and *surjective* from A to B.
   - f is an *identity* if it contains for every value a in its domain a tuple with first component a and also second component a. *f is the composition* of g and h if f contains a tuple $\langle a, b\rangle$ if and only if g contains a tuple $\langle a, c\rangle$ and h contains a tuple $\langle c, b\rangle$ for some value b. The *composition* of g and of h is that function f that satisfies above criterion.
   - The *image* under f of a set A is that set that contains every value b such that f contains a tuple with first component a and second component b for some value a in A. The *preimage* under f of a set B is that set that contains every value a such that f contains a tuple with first component a and second component b for some value b in B.