CSCI 189 Assignment 4 Problem Set

This assignment povides an opportunity to learn about sets and their properties. We will illustrate these fundamental ideas using examples from computer science.

The following table provides some useful information for the ideas below:

Name	*Class*	*Home*	*Calc*	*Prog*	*Career*	*Subject*	*Xwords*
Chris	2007	MA	no	yes	computers	cs	yes
Evan	2007	NC	hs	yes	unk	cs	no
Jarrett	2008	MD	yes	yes	computers	cs	yes
Jason	2010	VA	no	no	pilot	aeronautics	yes
Joe	2008	MA	yes	yes	banking	econ	yes
John	2008	NY	yes	yes	unk	cs	yes
Karina	2007	NY	yes	no	unk	anthro	no
Kate	2006	ME	hs	no	unk	english	yes
Matt	2009	MA	hs	no	business	unk	yes
Nolan	2008	CA	yes	yes	computers	cs	yes

Summary of Concepts in Section 3.1

A *set* is a collection of distinct elements taken from a particular domain. A set can be described either:

1. By enumerating its elements, in the form $S = \{s_1, s_2, \ldots, s_n\}$, or
2. By describing a property that all its members share, in the form $S = \{x \mid P(x)\}$

Here are a few important sets:

$N =$ the nonnegative integers $\{0, 1, 2, \ldots\}$ (defined in Haskell as `[0,1..]`),

$Z =$ the integers $\{\ldots -3, -2, -1, 0, 1, 2, 3, \ldots\}$ (known in Haskell as the type `Integer`),

$Q =$ the rational numbers (known in Haskell as the type `Rational`),

$R =$ the real numbers (similar, but not quite identical, to the Haskell types `Float` and `Double`),

$C =$ the complex numbers,

`Bool` = the Haskell type {`True, False`} of Boolean values (denoted by {T, F} in your text),

`Char` = the Haskell type that describes the set of all characters on the keyboard (sometimes called the ASCII set), and

L = a "language", or a set of all *strings* of characters from `Char` (known in Haskell as the type `String`).

$Names = \{Chris, Evan, Jarrett, Jason, Joe, John, Karina, Kate, Matt, Nolan\}$

$Class = \{2006, 2007, 2008, 2009, 2010\}$

$Home = \{CA, MA, MD, ME, NC, NY, VA\}$

$Calc = \{yes, no, hs\}$

$Prog = \{yes, no\}$

$Career = \{banking, business, computers, pilot, unk\}$

$Subject = \{aeronautics, anthro, cs, econ, english, unk\}$

$Xwords = \{yes, no\}$

Set Operations

Set *membership* is written as $x \in A$, and denotes that x is a member of set A. In Haskell, this is written as:

```
Asst4> member x a
```

The empty set, which has no members, is written as Φ, and in Haskell as `phi`.

Subset: $A \subseteq B$ (read "A is a subset of B") if every member of A is also a member of B. That is, $\forall x(x \in A \rightarrow x \in B)$. In Haskell, this is written `a <= b`.

Proper subset: $A \subset B$ if $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$. In Haskell, this is written `a < b`.

Power set: the set of *all* subsets of A. In Haskell, this is written `powerset a`.

Universal set: U = the domain (universe) of values from which the elements of a set are taken.

Intersection: $A \cap B = \{x \mid x \in A \wedge x \in B\}$. In Haskell, this is written `a /\ b`.

Union: $A \cup B = \{x \mid x \in A \vee x \in B\}$. In Haskell, this is written `a \/ b`.

Difference: $A - B = \{x \mid x \in A \wedge x \notin B\}$. In Haskell, this is written `a ~~ b`.

Complement: $A' = U - A$. In Haskell, this is written `u ~~ a`.

Product: $A \times B = \{(x, y) \mid x \in A \wedge y \in B\}$. In particular, $A \times A = A^2$ and $A \times A \times \ldots \times A = A^n = \{(x_1, x_2, \ldots, x_n) \mid \forall i(1 \leq i \leq n \rightarrow x_i \in A)\}$. In Haskell, the product is written `a * b`, and A^n is written `a ^ n`.

The Haskell definitions for these operations are found in the file `Set.lhs`, and a tutorial on using them is found in the file `Asst4.lhs`. This software will provide a useful tool as you solve the problems in this assignment.

Set Properties

The following properties apply to all sets.

Property	Meaning	
Commutativity	$A \cup B = B \cup A$	$A \cap B = B \cap A$
Associativity	$A \cup (B \cup C) = (A \cup B) \cup C$	$A \cap (B \cap C) = (A \cap B) \cap C$
Distributivity	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
Identity	$A \cup \emptyset = A$	$A \cap U = A$
Complement	$A \cup A' = U$	$A \cap A' = \emptyset$

The *cardinality* of a finite set $S = \{s_1, s_2, \ldots, s_n\}$ is n. A set is *countable* if its elements can be ordered in such a way that they can be counted (that is, associated with the integers 1, 2, ...). All finite sets, and some infinite sets, are countable (e.g., the set of rational numbers Q) and some are not (e.g., the set of real numbers R). The trick to determining countability is to see whether an arrangement of the elements can be found so that they can be counted (i.e., lined up with the positive integers).

Most sets encountered in computer science are countable (though some are very large!). For instance, the set of all strings of alphabetic characters on the keyboard is countable. That is, we can arrange this set for counting in the following way:

The empty string	`""`
The 26 strings of length 1	`"a" "b" "c" ... "z"`
The 26^2 strings of length 2	`"aa" "ab" ... "az" "ba" "bb" ... "zz"`
The 26^3 strings of length 3	`"aaa" "aab" ... "zzz"`
And so forth	

Problems to be handed in

Section 3.1 (p 178) Exercises 3, 5, 7ac, 10, 17hij, 21, 33bg, 40, 41, 44def, 47, 50ac, 55.

Please work individually on this assignment.