CS6610: Software Engineering II

Lecture 2: Sets and Relations
This Lecture...

- reviews the concepts of sets and relations required for Alloy.
- focuses on the forms of set operation and definitions used in specifications.
Set

- Collection of distinct objects
- Each set’s objects are drawn from a larger *domain* of objects all of which have the same type --- sets are homogeneous
- Examples:
  
  - \{2, 4, 5, 6, \ldots\}  
  - \{red, yellow, blue\}  
  - \{true, false\}  
  - \{red, true, 2\}  

  set of integers
  set of colors
  set of boolean values
  for us, not a set!
Value of a Set

- Is its membership
- Two sets $A$ and $B$ are equal if
  - every member of $A$ is a member of $B$
  - every member of $B$ is a member of $A$
- $x \in S$ denotes "$x$ is a member of $S$"
Defining Sets

- We can define a set by *enumeration*
  - PrimaryColors == \{red, yellow, blue\}
  - Boolean == \{true,false\}
  - Evens == \{..., -4, -2, 0, 2, 4, ...\}

- This works fine for finite sets, but
  - what do we mean by “...”?
  - remember we want to be precise
Defining Sets

- We can define a set by *comprehension*, that is, by describing a property that its elements must share.

**Notation:**
- \{ x : S | P(x) \}
- Form a new set of elements drawn from set/domain \( S \) including exactly the elements that satisfy predicate (i.e., boolean function) \( P \).

**Examples:**
- \{ x : N | x < 10 \} \quad \textit{Naturals less than 10}
- \{ x : Z | ( \exists y : Z | x = 2y) \} \quad \textit{Even integers}
- \{ x : N | \text{false} \} \quad \textit{Empty set of natural numbers}
Cardinality

- **Cardinality** is the size of a set
- Examples:
  - # \{red,yellow,blue\} = 3
  - # \{1,2,2\} = 2
  - # \text{Z}
- Don’t worry about infinite sets too much. We’ll be using them in a very practical way
Set Operations

- **Union:**
  - $X \cup Y \equiv \{e \mid e \in X \lor e \in Y\}$
  - $\{\text{red}\} \cup \{\text{blue}\} = \{\text{red, blue}\}$

- **Intersection**
  - $X \cap Y \equiv \{e \mid e \in X \land e \in Y\}$
  - $\{\text{red, blue}\} \cap \{\text{blue, yellow}\} = \{\text{blue}\}$

- **Difference**
  - $X \setminus Y \equiv \{e \mid e \in X \land e \notin Y\}$
  - $\{\text{red, yellow, blue}\} \setminus \{\text{blue, yellow}\} = \{\text{red}\}$
Subsets

- A **subset** holds elements drawn from another set
  - \( X \subseteq Y \equiv (\forall e \mid e \in X \rightarrow e \in Y) \)
  - \( \{1,7,17,23\} \subseteq \mathbb{Z} \)
- A **proper subset** is a non-equal subset
- Another view of equality
  - \( A = B \equiv (A \subseteq B \land B \subseteq A) \)
Power Sets

- The power set of set $S$ (denoted $\mathcal{P}(S)$) is the set of all subsets of $S$, i.e.,

$$\mathcal{P}(S) \equiv \{ e \mid e \subseteq S \}$$

- Example:
  - $\mathcal{P}({a,b,c}) = \{ \emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,c\} \}$

Note: for any $S$, $\emptyset \subseteq S$ and thus $\emptyset \in \mathcal{P}(S)$
For you to do

- Specifying using comprehension notation
  - Odd positive integers
  - The squares of integers, i.e, \{1, 4, 9, 16, \ldots\}
Set Partitioning

- Sets are *disjoint* if they share no elements.
- Often when modeling, we will take some set $S$ and divide its members into disjoint subsets called *partitions*.
- Each member of $S$ belongs to exactly one partition.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Soup</strong></td>
<td><strong>Chips &amp; Salsa</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Steak</strong></td>
<td><strong>Pizza</strong></td>
<td><strong>Sweet&amp;Sour Pork</strong></td>
</tr>
<tr>
<td><strong>Cake</strong></td>
<td><strong>Apple pie</strong></td>
<td><strong>Ice Cream</strong></td>
</tr>
</tbody>
</table>
Model residential scenarios

- Basic domains: *Persons*, *Residences*
- Partitions:
  - Partition *Persons* into *Child*, *Student*, *Adult*
  - Partition *Residences* into *Home*, *DormRoom*, *Apartment*, *Shelter*
For you to do

- Express the following properties of pairs of sets
  - Two sets $A$, $B$ are disjoint
  - Two sets $A$, $B$ form a partitioning of a third set $C$
Acknowledgements

- A significant portion of these slides are adapted from Matt Dwyer, John Hatcliff, and Rod Howell's slides of the course software specification