Basic Set Notation

- A set is an unordered collection of elements
  - Use “curly” brackets, e.g. {x, y, z} is a set containing three elements
  - {} or {∅} is the empty set
  - $x \in S$ indicates that x is an element of the set *S*

$$2 \in \{1, 2, 5\}, \qquad w \in \{x, y, z, w\}$$

  - $x \notin S$ indicates that x is not an element of the set S

$$3 \notin \{1, 2, 5\}, \qquad 3 \notin \{\}$$

- If *S* and *T* are both sets:
  - *S* is a subset of *T*, written $S \subseteq T$, provided that every element of *S* is also an element of *T*
  - Write $S \nsubseteq T$ to indicate that *S* is not a subset of *T*
  - Also, for every set *S*, $\emptyset \subseteq S, S \subseteq S$
  - The power set of S, written $\mathcal{P}(S)$, is the set containing all the subsets of S:

$$\mathcal{P}(\{1, 2, 5\}) = \{\emptyset, \{1\}, \{2\}, \{5\}, \{1,2\}, \{1,5\}, \{2,5\}, \{1,2,5\}\}$$

Basic Set Operations

Notion	Notation	Definition
Union	$S \cup T$	$S \cup T = \{x \| x \in S \text{ or } x \in T\}$
Intersection	$S \cap T$	$S \cap T = \{x \| x \in S \text{ and } x \in T\}$
Set Difference	$S - T$	$S - T = \{x \| x \in S \text{ and } x \notin T\}$
Cartesian Product	SxT	$SxT = \{(x, y) \| x \in S \text{ and } y \in T\}$

$$\{1,2,5\} \cup \{2,4\} = \{1,2,4,5\}$$

$$\{1,2,5\} \cap \{2,4\} = \{2\}$$

$$\{1,2,5\} - \{2,4\} = \{2\}$$

$$\{1,2,5\} x \{2,4\} = \{(1,2), (1,4), (2,2), (2,4), (5,2), (5,4)\}$$

Syntax and Logical Formulas

- Syntax
  - From the matt.might.net website: Backus-Naur Form (BNF) notation is a formal notation for encoding grammars intended for human consumption. Many programming languages, protocols or formats have a BNF description in their specification. The symbol ::= means "may expand into" and/or "may be replaced with".
  - Dr. Chin says BNF specifications provide a way to state syntactic rules precisely and unambiguously. *Keep in mind that these syntactic rules are not necessarily related to syntactic logic!*
  - Example 2.4 (Access Control textbook pg 16)

$$
\begin{aligned}
\textbf{BinNumber} ::= {} & \textbf{Bit BinNumber} \\
\rightsquigarrow {} & 1\ \textbf{BinNumber} \\
\rightsquigarrow {} & 1\ \textbf{Bit BinNumber} \\
\rightsquigarrow {} & 1\ \textbf{Bit Bit} \\
\rightsquigarrow {} & 10\ \textbf{Bit} \\
\rightsquigarrow {} & 101
\end{aligned}
$$

Syntax and Logical Formulas Cont.

- Principal expressions
  - Principals are the main actors in a system, e.g. people, processes, cryptographic keys, personal identification numbers (PINs), userid-password pairs, etc.
  - Principals may be simple or compound.
  - Define **Pname** to be the collection of all simple principal names. For example,

  $\boldsymbol{PName} = \{Alice, Bob, the\ key\ K_{Alice}, the\ PIN\ 1234\}$

  - A compound principal is an abstract entity denoting a combination of principals, e.g. "the access control system ***A*** requires Alice's PIN and biometrics including Alice's finger-prints and retina scan".
  - P&Q denotes the abstract principal "P in conjunction with Q"
  - P|Q denotes the abstract principal "P quoting Q"
    - & "binds more tightly" than | therefore, $Sally\ \&\ Ted\ |\ Uly\ is\ (Sally\ \&\ Ted)|Uly$
    - Both & and | are associative

More Syntax and Logical Formulas

- Propositional variables
  - Denoted by lower-case letters, e.g. p, q, r, write, rff
- Well-formed Formulas (wff)
  - Examples (Access Control textbook pg 20)

$$r$$

$$((\neg q \wedge r) \supset s)$$

$$(Jill\ says\ \left(r \supset (p \vee q)\right))$$

Kripke Structures Example 2.7

- Three children; Flo, Gil and Hal
- Looked after a babysitter who only lets them go outside if it is sunny and warm
- The weather is such that it can only be; sunny and warm, sunny but cool, or not sunny.

$$W_0 = \{sw, sc, ns\}$$

- Use the propositional variable g to represent "go outside"
- The interpretation function I_0 is:

$$I_0 : PropVar \longrightarrow \mathcal{P}(\{sw, sc, ns\})$$

- So, $I_0(g) = \{sw\}$
- Gil is the tallest and can see the thermometer. He can tell when it is sunny and warm. That is, he has "perfect" knowledge and:

$$J_0(Gil) = \{(sw, sw), (sc, sc), (ns, ns)\}$$

- Flo is shorter. She cannot see the outdoor thermometer. So:

$$J_0(Flo) = \{(sw, sw), (sw, sc), (sc, sw), (sc, sc), (ns, ns)\}$$

- Hal is too young to understand how it can be both sunny and cool. He believes if the sun is out it must be warm. So:

$$J_0(Hal) = \{(sw, sw), (sc, sw), (ns, ns)\}$$

- That produces the Kripke structure: $\mathcal{M} = \langle W_0, I_0, J_0 \rangle$

Composition of Relations Simple Example

$$R_1 = \{(4,a),(4,b),(5,c),(6,a),(6,c)\}$$
$$R_2 = \{(a,1),(a,n),(b,1),(b,m),(c,1),(c,m),(c,n)\}$$

Q: What is $R_1 \circ R_2$?

A: Start with the first pair in R_1, i.e. $(4,a)$ map a to pairs in R_2, $(a,1),(a,n)$ to get the first two elements $(4,1),(4,n)$ and keep going…

$$R_1 \circ R_2 = \{(4,1),(4,n),(4,m),(5,1),(5,m),(5,n),(6,1),(6,m),(6,n)\}$$

You should be able to do this now!

$$J(Andy) = \{(w_0,w_0),(w_0,w_2),(w_1,w_1),(w_2,w_1)\}$$
$$J(Stu) = \{(w_1,w_2)\}$$
$$J(Keri) = \{(w_0,w_2),(w_1,w_2),(w_2,w_2)\}$$

$$\begin{aligned}
&J(Keri\,|(Andy\,\&\,Stu)) \\
&= J(Keri) \circ J(Andy\,\&\,Stu), \\
&= J(Keri) \circ \big(J(Andy) \cup J(Stu)\big), \\
&= J(Keri) \circ \{(w_0,w_0),(w_0,w_2),(w_1,w_1),(w_2,w_1),(w_1,w_2)\}
\end{aligned}$$
$$= \{(w_0,w_1),(w_1,w_1),(w_2,w_1)\}$$

Do Exercise 2.3.1!

How about Evaluation Functions

Continue to use the information from Example 2.7 (pg 24)

Propositional Variables: (pg 29)

The truth of a propositional variable p is determined by the interpretation function I: a variable p is considered true in world w when $w \in I(p)$. Thus, for all propositional variables p,

$$E_{\mathcal{M}}[\![p]\!] = I(p)$$

For example, if $\mathcal{M}_0$ is the Kripke structure $\langle W_0, I_0, J_0 \rangle$ from Example 2.7, $E_{\mathcal{M}_0}[\![g]\!] = I_0(g) = \{sw\}$.

Negation:

A formula of the form $\neg\varphi$ is true in the worlds where φ is not true. Because (by definition) $E_{\mathcal{M}}[\![\varphi]\!]$ is the set of worlds in which φ is true, we define

$$E_{\mathcal{M}}[\![\varphi]\!] = W - E_{\mathcal{M}}[\![\varphi]\!]$$

Thus, returning to Example 2.7,

$$E_{\mathcal{M}}[\![\neg g]\!] = W_0 - E_{M_0}[\![g]\!] = \{sw, sc, ns\} - \{sw\} = \{sc, ns\}.$$

Notice that $E_{\mathcal{M}}[\![\neg g]\!]$ is the set of worlds in which the children are not allowed to go outside.

Kripke Semantics

Recall $\mathcal{M}_1 = \langle W_1, I_1, J_1 \rangle$ where $W_1 = \{x, y, t\}$ and $I_1(q) = \{x, t\}$, $I_1(r) = \{y\}$, $I_1(s) = \{y, t\}$

That is, the universe W_1 is given by the set $\{x, y, t\}$ and for the interpretation function I, q is true in worlds x, t; r is true in world y; and s is true in worlds y, t.

In what worlds is $q \supset (r \wedge s)$ true? (If q, then in what worlds is "r and s" true? or "q implies…")

$E_{\mathcal{M}}[\![q \supset (r \wedge s)]\!] =$???

We see that the top level operator is "implication". So, recall the definition of implication (pg 30):

$E_{\mathcal{M}}[\![\varphi_1 \supset \varphi_2]\!] = (W - E_{\mathcal{M}}[\![\varphi_1]\!]) \cup E_{\mathcal{M}}[\![\varphi_2]\!]$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = \left(W_1 - E_{\mathcal{M}_1}[\![q]\!]\right) \cup E_{\mathcal{M}_1}[\![(r \wedge s)]\!]$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = \left(W_1 - E_{\mathcal{M}_1}[\![q]\!]\right) \cup (E_{\mathcal{M}_1}[\![(r)]\!] \cap E_{\mathcal{M}_1}[\![(s)]\!])$

Replace the functions for r and s with the values where r and s are true.

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = (W_1 - I_1(q)) \cup (I_1(r) \cap I_1(s))$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = (\{x, y, t\} - \{x, t\}) \cup (\{y\} \cap \{y, t\})$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = \{y\} \cup \{y\}$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = \{y\} \cup \{y\}$

$E_{\mathcal{M}_1}[\![q \supset (r \wedge s)]\!] = \{y\}$

Access-Control Operators

Access-control operators of the logic (e.g. says, controls, and $\Rightarrow$ (speaks for)) have more interesting semantics (pg 31).

Says:

Controls:

Speaks For:

Example 2.14

Recall $\mathcal{M}_0 = \langle W_0, I_0, J_0 \rangle$ from Example 2.7. The set of worlds W_0 in which the formula $Hal\ says\ g\ is\ true$ is given by $E_{M_0}[\![Hal\ says\ g]\!]$, which is calculated as follows:

$$\begin{aligned} E_{M_0}[\![Hal\ says\ g]\!] &= \{w \mid J_0(Hal)(w) \subseteq E_{M_0}[\![g]\!]\} \\ &= \{w \mid J_0(Hal)(w) \subseteq \{sw\}\} \\ &= \{sw, sc\} \end{aligned}$$

This result captures Hal's mistaken belief that, whenever it is sunny (i.e. when the current world is either sw or sc), the children will be able to go outside.

In contrast, recall that Flo is unable to distinguish the two worlds sw and sc. Specifically, the relation $J_0(Flo)$ has the following properties:

$$\begin{aligned} J_0(Flo)(sw) &= \{sw, sc\}, \\ J_0(Flo)(sc) &= \{sw, sc\}, \\ J_0(Flo)(ns) &= \{ns\}. \end{aligned}$$

Thus, the worlds in which $Flo\ says\ g\ is\ true$ can be calculated as follows:

$$\begin{aligned} E_{M_0}[\![Flo\ says\ g]\!] &= \{w \mid J_0(Flo)(w) \subseteq E_{M_0}[\![g]\!]\} \\ &= \{w \mid J_0(Flo)(w) \subseteq \{sw\}\} \\ &= \emptyset. \end{aligned}$$

Moving Toward More Precision

- These informal meanings are helpful, but…
  - Which formulas are true in all cases (i.e. tautologies?)
  - Which formulas are never true (i.e. contradictions)?
  - Which formulas logically follow from others?
  - How can we reason precisely about these statements?
  - How can we trust that our conclusions make sense?
- To answer these questions the logic needs formal semantics, i.e. semantic logic:
  - Akin to truth tables for propositional logic
  - Based on **Kripke structures**

Terms of the HOL Logic

Kind of term	HOL *notation*	*Standard notation*	*Description*
Truth	`T`	$\top$	*true*
Falsity	`F`	$\bot$	*false*
Negation	`~`t	$\neg t$	*not t*
Disjunction	t_1 `\/` t_2	$t_1 \vee t_2$	t_1 *or* t_2
Conjunction	t_1 `/\` t_2	$t_1 \wedge t_2$	t_1 *and* t_2
Implication	t_1`==>`t_2	$t_1 \Rightarrow t_2$	t_1 *implies* t_2
Equality	t_1`=`t_2	$t_1 = t_2$	t_1 *equals* t_2
$\forall$-quantification	`!`x`.`t	$\forall x.\, t$	*for all* $x : t$
$\exists$-quantification	`?`x`.`t	$\exists x.\ t$	*for some* $x : t$
ε-term	`@`x`.`t	$\varepsilon x.\, t$	*an x such that:* t
Conditional	`if` t `then` t_1 `else` t_2	$(t \rightarrow t_1, t_2)$	*if t then* t_1 *else* t_2

Table 6.1: HOL Notation for Higher Order Logic Terms