Predicate logic

(Reading Gossett Section 2.6)

Predicates and Boolean Expressions

Recall that a **statement** or **proposition** is an assertion that may be labelled true or false

And that statements may depend on one or more variables.

Statements are also called **boolean expressions**.

Examples:

- $m = 3 \times n$
- $i < 0$

Predicate. A **predicate** is a function that results in a boolean value.

Examples: P and Q defined as follows are predicates

- $P(m, n) \triangleq (m = 3 \times n)$
- $Q(i) \triangleq (i < 0)$

Predicates and boolean expressions are often used in specification of systems or other objects. In this case, the variables represent observable quantities: E.g.

- $V = 100 \cdot I$
  - ∗ expresses a relationship between quantities identified as V and I
- $q(t + 1) = d(t)$, for all $t \in \mathbb{N}$.
  - ∗ expresses a relationship between functions identified as q and d
- $20 \le h \le 21 \wedge 10 \le w \le 11 \wedge 5 \le d \le 5.5$
  - ∗ expresses constraints on quantities identified as h, w, and d

Note that since the names connect the statements with physical quantities, the names are important.

Quantifiers

Informal definition

Review: Summation notation

$$\sum_{j \in S} f(j)$$

means (informally)

$$f(s_0) + f(s_1) + ...$$

where $S = \{s_0, s_1, ...\}$.

A common case is where S is of the form $\{m, m+1, ..., n\}$, in which case we write

$$\sum_{j=m}^{n} f(j)$$

For example, it is a theorem that

$$\left(\sum_{j \in \{1,...,n\}} j\right) = \frac{n(n+1)}{2}$$

New stuff. What if the operation is not $+$ but something else?

- If the operation is $\times$ then we use the notation
$$\left(\prod_{j \in S} f(j)\right) = f(s_0) \times f(s_1) \times ...$$
- If the operation is $\wedge$, then we use the notation
$$(\forall j \in S, f(j)) = f(s_0) \wedge f(s_1) \wedge ......$$
pronounced “for all”. We call this “universal quantification”. Of course $f(s_i)$ should be $\in \{F, T\}$ in each case.
- If the operation is $\vee$, then we use the notation
$$(\exists j \in S, f(j)) = f(s_0) \vee f(s_1) \vee ......$$
pronounced “there exists a”. We call this “existential quantification”

Examples

- "i is a multiple of 3"
  - ∗ We can rephrase this as

    "there exists an integer, k, such that $i = 3k$"

    or using our notation
    $$\exists k \in \mathbb{Z}, (i = 3k)$$
    this is the same as the infinite expression
    $$(i = 3 \cdot 0) \lor (i = 3 \cdot 1) \lor (i = 3 \cdot -1) \lor \text{ and so on}$$
  - ∗ Note that this is a conditional statement, as its truth depends on the value given to the variable i.
- "At least one of $100, 101,$ and 103 is a multiple of 3"
  $$\exists j \in \{100, 101, 103\}, \exists k \in \mathbb{Z}, (j = 3k)$$
  this is the same as
  $$\begin{array}{rl} & (\exists k \in \mathbb{Z}, (100 = 3k)) \\ \lor & (\exists k \in \mathbb{Z}, (101 = 3k)) \\ \lor & (\exists k \in \mathbb{Z}, (102 = 3k)) \end{array}$$
  - ∗ Note that this is a tautology. Both variables j and k are "local" to the expression, so the truth of it does not depend on their values.

Discrete Math. for Engineering, 2004. Notes 3. Theodore Norvell, Memorial University

- "For all integers, either the integer or the next integer or the one after that is divisible by 3"
  $$\forall i \in \mathbb{Z}, \exists j \in \{i, i + 1, i + 2\}, \exists k \in \mathbb{Z}, (j = 3k)$$
  - ∗ This too is a tautology.
- Suppose that P is a predicate on the integers. I.e. $P(j)$ is true if j has property P.
  - ∗ "There is no number with property P larger than j"
  - ∗ We can say for all larger numbers, P does not hold
    $$\forall k \in \mathbb{Z}, (k > j \to \neg P(k))$$
  - ∗ Or we could say that there does not exist a number larger than k for which P holds
    $$\neg \exists k \in \mathbb{Z}, (k > j \land P(k))$$
  - ∗ This is a conditional statement. Its truth depends on the what the property is and, perhaps, on what value j has.
- There is no largest prime. Let $prime$ be the property of being prime.
  - ∗ We can say that "for each integer there is a larger prime"
    $$\forall j \in \mathbb{Z}, \exists k \in \mathbb{Z}, (k > j \land prime(k))$$

- Suppose that y and x are functions of time in seconds.
  - “The value of y is always twice the value of x”
    $$\forall t \in \mathbb{R}, (y(t) = 2 \times x(t))$$
  - “The value of y is always the value of x delayed by 3 seconds”
    $$\forall t \in \mathbb{R}, (y(t) = x(t-3))$$
    or equivalently
    $$\forall t \in \mathbb{R}, (y(t+3) = x(t))$$
  - “The value of y is the value of x delayed by less than three seconds.
    $$\exists d \in (0,3), \forall t \in \mathbb{R}, (y(t+d) = x(t))$$

De Morgan’s law for quantifiers:

Recall that I claimed the expressions
$$\neg\forall k \in \mathbb{Z}, (k > j \rightarrow \neg P(k))$$
and
$$\exists k \in \mathbb{Z}, (k > j \wedge P(k))$$
are equivalent.

In general $\neg(\forall j \in S, f(j))$ will be equivalent to $(\exists j \in S, \neg f(j))$.

Informal proof: Consider $\neg(\forall j \in S, f(j))$. Where $S = \{s_0, s_1, ...\}$ We can find an equivalent expression as follows

$$\begin{aligned} & \neg(\forall j \in S, f(j)) \\ \Leftrightarrow\ & \neg(f(s_0) \wedge f(s_1) \wedge ...) \text{ Informal definition of } \forall \\ \Leftrightarrow\ & (\neg f(s_0) \vee \neg f(s_1) \vee ...) \text{ DeMorgan's law} \\ \Leftrightarrow\ & (\exists j \in S, \neg f(j)) \text{ Informal definition of } \exists \end{aligned}$$

So we have

DeMorgan’s law $\neg(\forall j \in S, f(j)) \Leftrightarrow (\exists j \in S, \neg f(j))$

From this we can prove

$$
\begin{aligned}
& \neg\left(\exists j \in S, f(j)\right) \\
\Leftrightarrow\ & \neg\left(\exists j \in S, \neg\neg f(j)\right) \text{ Involution} \\
\Leftrightarrow\ & \neg\neg\left(\forall j \in S, \neg f(j)\right) \text{ DeMorgan's law} \\
\Leftrightarrow\ & \left(\forall j \in S, \neg f(j)\right) \text{ Involution}
\end{aligned}
$$

So

$$\neg\left(\exists j \in S, f(j)\right) \Leftrightarrow \left(\forall j \in S, \neg f(j)\right)$$

Formally defining the quantifiers

We might formally define the quantifiers in terms of set notation.

$$
\begin{aligned}
\left(\exists x \in S, f(x)\right) &\Leftrightarrow \left(\{x \mid x \in S \wedge f(x)\} \neq \emptyset\right) \\
\left(\forall x \in S, f(x)\right) &\Leftrightarrow \left(\{x \mid x \in S \wedge f(x)\} = S\right)
\end{aligned}
$$

Using these definitions we can more formally prove laws. For example, here is a formal proof of the De Morgan's law we informally proved above. In this proof, we take the universe to be S.

$$
\begin{aligned}
& \neg\left(\forall j \in S, f(j)\right) \\
\Leftrightarrow\ & \neg\left(\{x \mid f(x)\} = S\right) \\
\Leftrightarrow\ & \left(\{x \mid f(x)\} \neq S\right) \\
\Leftrightarrow\ & \left(\overline{\{x \mid f(x)\}} \neq \emptyset\right) \\
\Leftrightarrow\ & \left(\{x \mid \neg f(x)\} \neq \emptyset\right) \\
\Leftrightarrow\ & \left(\exists j \in S, \neg f(j)\right)
\end{aligned}
$$

Free and bound variables

Whenever a variable occurs in an expression, we can classify the occurrence as either "free" or "bound".

- *Free occurrences* refer to quantities that are external to the expression.
- *Bound occurrences* are occurrences of variables that are local to the expression.

Consider an expression

$$V = 100 \times I$$

this expresses a relationship between external quantities named V and I. The names connect the expression to quantities in the real world. The occurrences of V and I are *free*.

Boolean expressions with free occurrences of variables can serve as describe relationships between physical quantities. The variable names serve to connect the expression to specific quantities in a design.

Bound occurrences of variables represent quantities that are entirely local to the expression. For example the expression

$$m = \left(\sum_{j \in \{1,\ldots,n\}} j \right)$$

expresses a relationship between numbers m and n, but the variable j is entirely local to the expression. We can rename variable j to something else without changing the meaning of the expression at all. For example

$$m = \left(\sum_{k \in \{1,\ldots,n\}} k \right)$$

Some other examples of bound variables.

- The expression

  $$w < \int_0^z e^x \, dx$$

  expresses a constraint on the values of w and z. All occurrences of x are bound. We could equivalently write

  $$w < \int_0^z e^y \, dy$$

- The expression
$$S = \{k \mid 0 \le k \le n^2\}$$
is equivalent to
$$S = \{j \mid 0 \le j \le n^2\}$$
Both express a relationship between S and n. The occurrences of k and j are bound.
- The expression
$$\forall t \in \mathbb{R}, (y(t+3) = x(t))$$
is equivalent to the expression
$$\forall u \in \mathbb{R}, (y(u+3) = x(u))$$
Both express a relationship between functions named x and y. The occurrences of t and u are bound.
- When we define a function by an equation, the parameter is a bound variable. For example, if we define a function f by the equation
$$f(t) = 2x(t-1)$$
the occurrences of t are considered bound. Such a definition is perhaps more properly written as
$$\forall t \in \mathbb{R}, (f(t) = 2x(t-1))$$
However, mathematicians and engineers habitually omit to write the quantification in such cases.

Important sanity check. If you are trying to express a constraint between named quantities. Make sure that the variables that occur free in the expression are the same as the names of the quantities you are trying to express.

For example.

Q. Express the constraint "x is a multiple of y"
A0. $x = ky$
A1. $\forall x \in \mathbb{Z}, \forall y \in \mathbb{Z}, \exists k \in \mathbb{Z}, (x = ky)$
A2. $\exists k \in \mathbb{Z}, (x = ky)$

In A0 we are expressing a relationship between 3 quantities, not 2.

In A1 we are expressing a property of the integers (a property that happens to be false). The expression in A1 is equivalent to F and does not express a constraint on x and y.

In A2 we get it right. The free variables are x and y, as one would expect.

Revisiting some definitions

Earlier we defined equivalence of propositional expressions. In a propositional expression, all occurrences are free and all variables are propositional variables. We don't need these restrictions.

Equivalence revisited. We say that any two expressions A and B are **equivalent** (written $A \Leftrightarrow B$) iff they are equal for all values of their free variables.

Note that A and B don't need to be boolean expressions.

For example

- $\exists k \in \mathbb{Z}, (x = ky)$ is equivalent to $\neg\forall j \in \mathbb{Z}, (x \neq jy)$
- $\exists k \in \mathbb{Z}, (x = ky)$ is *not* equivalent to $\exists k \in \mathbb{Z}, (w = kz)$

Tautology revisited. We say that a boolean expression is a **tautology**, if it is true for all values of its free variables.

For example, if we understand the type of x to be $\mathbb{Z}$ then

$$\exists k \in \mathbb{Z}, k > x$$

is a tautology. While it "formally" expresses a constraint on variable x, this constraint turns out to be satisfied for every integer value x.

Inference revisited. For boolean expressions A and B we can infer B from A (written $A \Rightarrow B$) iff $A \rightarrow B$ is a tautology.