4 PREDICATES

In the last chapter we looked at the integer type. We reviewed the arithmetic operators +, - and *, div and *mod*. We introduced *max* and *min*. We looked at the numerical comparison operators <, ⩽, > and ⩾. And we looked at rules of precedence for the arithmetic operators.

Now we see how the values of variables may be constrained by predicates and how satisfying a predicate defines a set. The types of predicate we shall discuss include:

$=, \in$	equals and membership
$<, >$	relations: less than and more than
$\wedge, \vee$	connectives: conjunction (and) and disjunction (or)

4.1 PREDICATES

A declaration introduces a variable. For example

$$n : \mathbb{Z}$$

introduces the variable named *n*, of type ℤ, without any *constraints* or *bounds* on the values that *n* may have.

Suppose we require *n* to be not less than one, values 1, 2, 3, ... are what we require. To declare that n **is** more than zero we would write

$$n > 0$$

$n > 0$ is an example of a predicate. Values 1, 2, 3, ... and so on are said to *satisfy the* predicate $n > 0$, i.e. make $n > 0$ true.

Putting the declaration and predicate together, as shown below, specifies a set.

$$\{\ n : \mathbb{Z} \mid n > 0\ \}$$

The declaration and predicate parts are separated by a | symbol.

The declaration to the left of the | is the *source* of the elements of the set.

The predicate to the right of the | is like a *filter*: only elements whose values are more than zero pass through the filter.

The set just defined above is the set of all integers greater than zero, that is, the set of natural counting numbers from one upwards.

$\{ \ n : \mathbb{Z} \ | \ n > 0 \ \} = \{ \ 1, 2, 3, ... \ \}$ [Note: the ... is not Z. It means and so on. We write it to help our understanding.]

We define a predicate in terms of the set of objects that satisfy it.

Look at this example.

$\{ \ n : \mathbb{Z} \ | \ n \in 1..5 \ \} = \{ \ 1, 2, 3, 4, 5 \ \}$

The declaration $n : \mathbb{Z}$ is the source. The predicate $n \in 1..5$ is the filter. It says *n* **is** a member of the set of integers from 1 up to 5 inclusive. Only elements from $\mathbb{Z}$ that satisfy the predicate pass through the filter .

A predicate puts a constraint on the values a variable might have. For example

capacity : $\mathbb{Z}$
capacity = 10

says *capacity* **is** 10.

We could use the declaration and predicate together to define the set containing just one integer element, 10.

$\{ \ n : \mathbb{Z} \ | \ n = 10 \ \} = \{ \ 10 \ \}$

EXERCISE 4.1

Using set display notation write out the contents of the sets defined below.

1 $\{ n : \mathbb{Z} \mid n = 7 \}$

2 $\{ n : \mathbb{Z} \mid n \in 1..3 \}$

3 $\{ h : \mathbb{N} \mid h < 6 \}$

4.2 THE CONNECTIVES

The connectives include *conjunction* (and) and *disjunction* (or). They allow us to connect smaller predicates together to form larger ones.

For example, given the declaration $n : \mathbb{Z}$, here are two small predicates that say n **is** more than zero and n **is** less than 6.

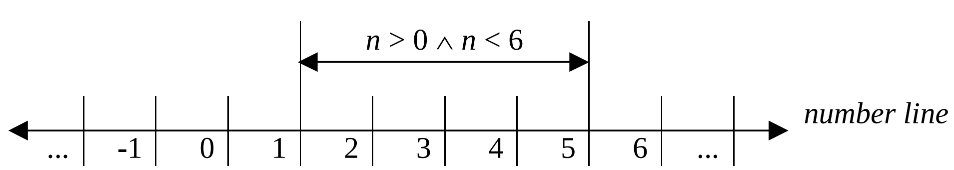
$n > 0$
$n < 6$

Now join them together with a $\wedge$, which means and-at-the-same-time.

$$n > 0 \wedge n < 6$$

We then have a *composite predicate* known as a conjunction. This predicate **is** true if n **is** more than 0 and, at the same time, n **is** less than 6.

$$\{ n : \mathbb{Z} \mid n > 0 \wedge n < 6 \} = \{ 1, 2, 3, 4, 5 \}$$



Here are two small predicates that say *n* **is** less than 1 and *n* **is** more than 5.

$n < 1$
$n > 5$

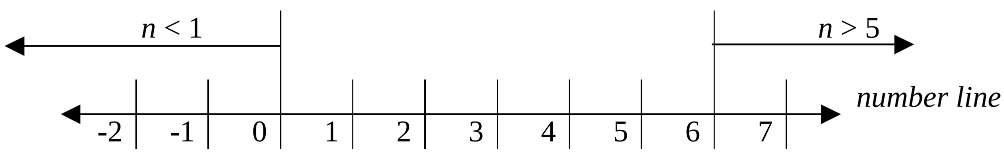
Now join them together with ∨, meaning or.

$n < 1 \vee n > 5$

We then have a composite predicate known as a *disjunction*. The predicate **is** true if either *n* **is** less than 1 or *n* **is** more than 5.

$\{ n : \mathbb{Z} \mid n < 1 \vee n > 5 \} = \{ ..., -2, -1, 0, 6, 7, 8, ... \}$

This is the set of all integers excluding { 1, 2, 3, 4, 5 }



It is easy to remember what the two symbols ∧ and ∨ mean when you notice that the ∧ looks a bit like the A in And.

EXERCISE 4.2

Using set display notation enumerate (list) the contents of the sets defined below.

1 $\{ n : \mathbb{Z} \mid n \geqslant -1 \wedge n \leqslant 1 \}$

2 $\{ d : \mathbb{Z} \mid d \geqslant 1 \wedge d \leqslant 7 \}$

3 $\{ n : \mathbb{Z} \mid n = 0 \vee n = 1 \}$

4 $\{ n : \mathbb{Z} \mid n > 0 \wedge n < 10 \wedge n \bmod 2 = 0 \}$

5 $\{ n : \mathbb{Z} \mid n > 0 \wedge n \operatorname{div} 5 < 1 \}$

REVIEW

We saw how the values of variables may be constrained by predicates. We saw how to define a predicate in terms of the set of objects that satisfy it. We discussed predicates including

$=, \in$	equals and membership
$<, >$	the relations less than and more than
$\wedge, \vee$	the connectives conjunction and disjunction

Next we see how schemas describe computer systems.

BIBLIOGRAPHY

JACKY J. 1997 *The Way of Z* Cambridge University Press pp 63, 66, 74, 96
SPIVEY J.M. 1992 *The Z Notation* Prentice Hall pp 28, 67
WOODCOCK J. & DAVIES J *Using Z: Specification, Refinement and Proof* Prentice Hall pp 28, 81