CHAPTER 3

Sets and types

Aims

To introduce some concepts of basic set theory, the Z notation for sets and types, and the idea of modelling systems using discrete mathematical structures.

Learning objectives

When you have completed this chapter, you should be able to:

- declare atomic and set-valued objects in Z;
- use logic to create set expressions;
- use sets and set operations to model the state of simple systems;
- understand the importance of types in Z.

3.1 Definitions

A *set* is a collection of distinct objects called *elements* or *members*. For example, the set of all people, or the set of all teapots. Every expression in a Z specification belongs to a set called its *type*, and whenever we introduce a new variable, we must declare its type. The type $\mathbb{Z}$ is the set of all integers, that is all whole numbers, and is built into the language. $\mathbb{N}$ is the set of all natural numbers and $\mathbb{N}_1$ is the set of all natural numbers not including zero.

$$\mathbb{N} = 0, 1, 2, \ldots$$
$$\mathbb{N}_1 = 1, 2, 3, \ldots$$

We may also introduce our own so-called *basic types* or *given sets*, by giving the singular name of the required type, in capitals, in square brackets, with an accompanying explanation. For example,

$[PERSON]$ the set of all people

Note that no indication is given as to how individual members of the set are represented.

Another way of introducing a new type is by enumerating the names of the elements of the type in a *free type definition*. For example,

$$COLOUR ::= red \mid green \mid blue$$
$$FUEL ::= petrol \mid diesel \mid electricity$$

We introduce *variables*, that is names denoting values from the above types, by writing *declarations*. For example,

$$p : PERSON$$
$$crayon : COLOUR$$
$$powerSource : FUEL$$

A collection of declarations such as this is called a *signature*. *p*, *crayon* and *powerSource* are now the names of variables which may be associated with any single member of their respective types. To declare more than one variable of the same type, we can use a shorthand form as follows:

$$p, q, r : PERSON$$
$$myFavourite, yourFavourite : COLOUR$$

We can test the value of a variable by propositions such as

$$crayon = green$$

3.2 Ways of describing sets

We may describe sets informally by stating a property common to objects which are elements of the set, for example

'the set of all whole numbers which are greater than 3 and less than 10'

In Z, we may describe this set *in extension*, that is we enumerate the elements between curly brackets, separated by commas

$$numset == \{4, 5, 6, 7, 8, 9\}$$

The $==$ sign denotes an *abbreviation definition*, which is used to introduce a global constant into a Z specification. The identifier *numset* becomes a name for the constant value $\{4, 5, 6, 7, 8, 9\}$. Note the difference between this and

equality (=), which is a means of constructing a predicate from two expressions of the same type.

For sets which are integer subranges such as this, we may use the equivalent notation

$$numset == 4 \,..\, 9$$

Sets may also be described by a *set comprehension*, whereby we introduce a predicate which characterises members of the set. The form of a set comprehension is

$$\{\text{declaration} \mid \text{predicate} \bullet \text{expression}\}$$

The declaration introduces one or more bound variables, the values of which are then constrained by the predicate. The form of the elements of the set is then given by the expression. For example, the above set may be described by

$$numset == \{n : \mathbb{Z} \mid n \geqslant 4 \land n \leqslant 9 \bullet n\}$$

read as '*numset* is a set defined by declaring an integer and constraining its possible values to be greater than or equal to 4 and less than or equal to 9; the elements of numset are precisely these values'.

If there is only one variable in the declaration, and the final expression is that variable, then the latter may be omitted, so the above example may be written

$$numset == \{n : \mathbb{Z} \mid n \geqslant 4 \land n \leqslant 9\}$$

The predicate may be omitted if it is always true, for example

$$evenints == \{n : \mathbb{Z} \bullet 2 * n\} \qquad \text{the set of all even integers}$$

The empty set, that is the set with no members, is represented by $\varnothing$ or $\{\}$.

Strictly, types in the Z language must be *maximal*, that is they must not be subsets of any other sets in the specification in which they occur, and therefore $\mathbb{N}$ and $\mathbb{N}_1$ are not actually types but subsets of the type $\mathbb{Z}$. If they were not available in the Z language, we could define them as follows:

$$\mathbb{N} == \{n : \mathbb{Z} \mid n \geqslant 0\}$$
$$\mathbb{N}_1 == \{n : \mathbb{Z} \mid n > 1\}$$

Thus, in a declaration such as $x : \mathbb{N}$, the type of x is actually $\mathbb{Z}$, but with the implicit constraint that x can only take non-negative values.

Exercises 3.1

1. Define the following sets in extension:
   (i) $\{x : \mathbb{N} \mid x < 4 \bullet 3 * x\}$
   (ii) $\{x, y : \mathbb{N} \mid x < 6 \land y < x \bullet x - y\}$
   (iii) $\{x, y : \mathbb{N}_1 \mid x < 5 \land y < 5 \bullet x + y\}$
   (iv) $\{x, k : \mathbb{N} \mid x = k^2 \land x \leqslant 10 \bullet x\}$
2. Define the following sets as comprehensions:
   (i) $\{1, 2, 3\}$
   (ii) $\{0, 1, 4, 9, 16\}$
   (iii) $\{0, 2, 6, 12\}$

3.3 Set operations

Membership

The set membership operator $\in$ is used to test whether an object is a member of a set. The expression $n \in S$ is read as 'n is a member of set S'. For the expression to be valid, n must be of the same type as the elements of S. The following are all true:

$$green \in COLOUR$$
$$4 \in \mathbb{Z}$$
$$diesel \in FUEL$$

Non-membership is tested using $\notin$. The following are all true:

$$green \notin \{red, blue\}$$
$$4 \notin \{\}$$
$$6 \notin \{x : \mathbb{Z} \mid x \geqslant 0 \land x \leqslant 10 \bullet x * 2 + 1\}$$

Exercise 3.2

Which of the following expressions are valid, and, for the valid ones, which are true and which are false?

(i) $4 \in \{4\}$ ✓ T
(ii) $\forall x : \mathbb{Z} \bullet x \in \mathbb{N}_1 \Rightarrow x \in \mathbb{N}$ ✓ F
(iii) $diesel \in \{5, 6\}$ ×
(iv) $2 \in \{x : \mathbb{Z} \mid x \in \mathbb{N} \land x < 0\}$ ✓ F

(v) $\{4\} \in \{4, 5\}$
(vi) $\{x : \mathbb{N} \mid x + 3 = 6 \bullet 2 * x\} \in \{\{4, 5\}, \{6\}\}$
(vii) $\{\} \in \{\}$
(viii) $\{1\} \notin \{\{\{1\}\}\}$

Cardinality

The number of elements in a set S is called its *cardinality*, denoted by $\#S$. For example,

$$\#\{2, 4, 6\} = 3$$

The cardinality of the empty set is zero:

$$\#\{\} = 0$$

Set equality

Two sets are equal iff they contain exactly the same members. The following expressions are all true:

$$\{1, 2, 3\} = \{2, 3, 1\}$$
$$\{1, 2, 3\} = \{1, 1, 1, 1, 1, 2, 3\}$$
$$\{\} = \{x : \mathbb{Z} \mid x > 2 \wedge x < 2\}$$

Note that repeated elements are not significant, as an element can only be in a set once. Also, the order in which the elements are written is not significant.

Subset

For any given sets S, T the expression

$$S \subseteq T$$

is read as 'S is a subset of T', and is a predicate which is true iff every member of S is a member of T. The following expressions are all true:

$$\{1, 2\} \subseteq \{1, 2, 3\}$$
$$\{\} \subseteq \{1, 2, 3\}$$
$$\{1, 2, 3\} \subseteq \{1, 2, 3\}$$
$$\{\} \subseteq \{diesel\}$$
$$\{\} \subseteq \{\}$$

Note that the empty set is a subset of every set, and every set is a subset of itself. We can test whether a set S is a so-called *proper subset* of a set T, that is

$$S \subseteq T \wedge S \neq T$$

using the predicate

$$S \subset T$$

Exercises 3.3

1. What are the values of the following expressions?

(i) $\#\{1, 2, 3\}$
(ii) $\#\{1, 1, 1\}$
(iii) $\#\{\{1, 2\}, \{\}, \{3\}\}$
(iv) $\#\{\{\{1\}\}\}$
(v) $\#\{\}$
(vi) $\#\{\{\}\}$

2. What are the values of the following expressions?

(i) $\{x, y : \mathbb{N} \mid x + y = 5 \bullet 2 * x\} = \{0, 2, 4, 6, 8, 10\}$
(ii) $\{red\} \subset \{red, blue\}$
(iii) $\{1, 2, 3\} \subset \{3, 1, 2\}$
(iv) $\{\} \subseteq \{\{\{1\}, \{2\}\}\}$

Powerset

The powerset of a set S, denoted by $\mathbb{P}S$, is the set of all the subsets of S. For example,

$$\mathbb{P}\{1, 2, 3\} = \{\{\}, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

If S is a type, then so is $\mathbb{P}S$. Variables of type $\mathbb{P}S$ take values which are *sets*. Note that

$$\#\mathbb{P}S = 2^{\#S}$$

so in the above example we have $\#\mathbb{P}\{1, 2, 3\} = 2^3 = 8$.

For example, the declaration

$$x : \mathbb{Z}$$

states that x is a variable which can represent integers, while the declaration

$$y : \mathbb{P}\mathbb{Z}$$

states that y is a variable which can represent *sets* of integers. Typical values might be $x = 5$ and $y = \{1, 2, 3\}$.

We can introduce a variable of any type T by a declaration involving not T but an expression whose type is $\mathbb{P}T$; that is, whose value is a subset of T. It is always possible in such declarations to infer the type of the variable from that of the expression. For example, given the definition

$$numset == \{4, 5, 6, 7, 8, 9\}$$

the declaration

$$x : numset$$

simultaneously introduces a variable of type $\mathbb{Z}$ and constrains its value to the set *numset*. Similarly the declaration

$$x : \mathbb{N}$$

simultaneously introduces a variable of type $\mathbb{Z}$ and constrains its value to the set $\mathbb{N}$.

The type of an empty set can usually be inferred from the context in which it appears. For example, in the expression $\{\} \in \{\{1\}, \{2,3\}, \{\}\}$ the type of the empty sets is $\mathbb{P}\mathbb{Z}$.

Exercises 3.4

1. Given the declaration $x : \mathbb{P}(\mathbb{P}\mathbb{Z})$, write down a typical value which may be associated with the variable x.
2. What is the type and cardinality of the following sets?
   (i) $\{\{1\}, \{2,3\}\}$
   (ii) $\{\{\{1\}\}, \{\{2,3\}\}, \{\}\}$
   (iii) $\{x : \mathbb{P}\mathbb{N} \mid x \in \mathbb{P}\{1,2,3\} \wedge x \subseteq \{1,2\}\}$

Union

The union of two sets S and T, written as

$$S \cup T$$

is the set consisting of all the members from S and T. For the expression to be valid, the sets S and T must be of the same type. For example,

$$\{1, 2, 3\} \cup \{3, 4, 5\} = \{1, 2, 3, 4, 5\}$$

As a further example, if the set of students at a given university is [*STUDENT*], we may define the following sets:

skiing: $\mathbb{P}$ *STUDENT*	the set of all students in the university skiing club
badminton: $\mathbb{P}$ *STUDENT*	the set of all students in the university badminton club

Then the set of all students who are in the skiing club, the badminton club, or both, is

$$skiing \cup badminton$$

Intersection

The intersection of two sets S and T, written as

$$S \cap T$$

is the set consisting of all the members which are in both S and T. Again, for the expression to be valid, the sets S and T must be of the same type. For example,

$$\{1, 2, 3\} \cap \{3, 4, 5\} = \{3\}$$

From the above example, the set of all students who are in both the skiing club *and* the badminton club is

$$skiing \cap badminton$$

Generalised union and intersection

Generalised union and intersection are special versions of the above operators, which are applied to a set of sets. The *generalised union* of a set of sets S, written as

$$\bigcup S$$

is the set comprising all elements which are members of at least one of the members of S. For example,

$$\bigcup\{\{1,2\},\{2,3,4\},\{1,2,4,5\}\} = \{1,2,3,4,5\}$$

The *generalised intersection* of a set of sets S, written as

$$\bigcap S$$

is the set comprising those elements which are members of all the members of S. For example,

$$\bigcap\{\{1,2\},\{2,3,4\},\{1,2,4,5\}\} = \{2\}$$

As a further example, if the university also has the following clubs

$$football, drama, yoga : \mathbb{P}\ STUDENT$$

then the set of those students who are members of at least one of the clubs is

$$\bigcup\{skiing, badminton, football, drama, yoga\}$$

and the set of those students who are members of all of the clubs is

$$\bigcap\{skiing, badminton, football, drama, yoga\}$$

Set difference

The difference of two sets S and T, written as

$$S \setminus T$$

is the set consisting of all the members of S which are not in T. For example,

$$\{1,2,3,4\}\setminus\{1,3,6,8\} = \{2,4\}$$

As a further example, the set of all students who are in the skiing club and not in the badminton club is

$$skiing \setminus badminton$$

Exercises 3.5

1. State whether each of the following expressions is valid, and simplify those which are:
   (i) $(\{1,2,3\}\cup\{2,3,4\})\cap\{5\}$ = $\emptyset$
   (ii) $\{1,2,3\}\setminus 4$ ×
   (iii) $\{1,2,3\}\setminus\{2,1,3\}$ = $\emptyset$
   (iv) $\{1,2,3\}\cap(\{1,2,3\}\cup\{\{1\},\{2\},\{3\}\})$ ×
2. What is the value of the expression $4 \in (\{4,5\}\setminus\{4,2\})$? F
3. Given the following,

$$skiing = \{tony, fred, alice, sarah, diana, susan, bill, henry, don\}$$
$$badminton = \{colin, sarah, fred, carol, don\}$$
$$football = \{bill, colin, alice, don, tony\}$$
$$drama = \{sarah, don\}$$
$$yoga = \{henry, don, carol, alice\}$$

simplify the following expressions:

(i) $(badminton \cap football)\setminus skiing$
(ii) $\{x : \mid x \in \mathbb{P}\ skiing \wedge ((x \cap drama) \subseteq badminton)\}$
(iii) $\bigcup\{skiing, badminton, drama\}$
(iv) $\bigcap\{skiing, badminton, football, drama, yoga\}$
(v) $\bigcap \mathbb{P}\ skiing$
(vi) $\bigcup \mathbb{P}\ skiing$
(vii) $\mathbb{P}(\bigcap\{skiing, badminton\})$

3.4 Sets and logic

Before leaving our introduction to sets, it is worth repeating that both elementary set theory and propositional logic are examples of *boolean algebras*. By making correspondences between set operators and logic operators, we arrive at the same five basic laws of boolean algebra which were discussed in Chapter 2. This is important, because it means that every proven logic theorem has an equivalent theorem within set theory, and results proved in one system are valid in the other. However, further discussion of this is beyond the scope of this book.

A major claim for formal specification languages is that they enable the construction of precise, unambiguous statements of requirements. In Z, the languages of sets and logic are the basic tools used to make such statements.

However, you will note that the set operators introduced in this chapter have been described in English. We could have given precise definitions using logic. For example, the following is a definition of set union. For a given type T

$$\forall x : T; A, B : \mathbb{P}T \bullet x \in A \cup B \Leftrightarrow x \in A \vee x \in B$$

In other words

$$\forall A, B : \mathbb{P}T \bullet A \cup B = \{x : T \mid x \in A \vee x \in B\}$$

Equality used as a predicate

Exercises 3.6

1. Give logic expressions to define formally the meaning of:
   (i) set intersection;
   (ii) set difference;
   (iii) generalised union.
2. Describe the following situation using the notation covered in this chapter. Assume that you have the type [*PERSON*], the set of all people.
   (i) People are either women or men, but not both.
   (ii) A company employs people in three departments: marketing, personnel and production. Each employee is in precisely one of these departments.
   (iii) Each department has a maximum of 10 staff.
   (iv) All the staff in marketing are women.
   (v) The company employs more men than women.
3. Now assume that each employee in the previous question can be in more than one department. Write down expressions for:
   (i) The number of women who work in all three departments.
   (ii) The number of men who work in marketing and personnel but not in production.

CHAPTER 4

The structure of a Z specification

Aims

To apply the material of the previous two chapters to the development of Z specifications, to introduce mechanisms for structuring Z specifications, and to illustrate the above with a simple example.

Learning objectives

When you have completed this chapter, you should be able to:

- represent the state of simple systems using sets and logic;
- specify the effect of operations which change or interrogate the state of a system;
- structure your specifications using schemas;
- introduce appropriate exception handling to totalise your operations using the schema calculus;
- determine the conditions necessary for an operation to take place successfully.

4.1 Introduction

In this chapter, we will develop an example to illustrate the 'flavour' of writing specifications with Z. We will introduce most of the notation for structuring Z specifications that will be used for the rest of the book. The specification is for the student badminton club mentioned in Chapter 3, and could be implemented as a computer system or paper records, to keep track of the whereabouts of the club members, add or remove members from the club, etc.

To write a specification in Z, we create a *model* of the required system. The structure, or state, of the system is represented using sets, and the relationships