B Exercises 1
Solutions to Sets

1. Given $X = \{a, b\}$, show the following.

   (a) $\mathsf{card}(X)$
   Ans: 2. There are two elements, a and b.

   (b) if $aa : X$, **one** possible value of aa
   Ans: Either of a or b. **Not** $\{\}$ or $\{a, b\}$!

   (c) $\mathbb{P}(X)$, $\mathsf{card}(\mathbb{P}(X))$.
   Ans: $\{\{\}, \{a\}, \{b\}, \{a, b\}\}$
   (The set of elements taken none at a time, one at a time, and two at a time.)
   Ans: 4

   (d) if $aaa : \mathbb{P}(X)$, **one** possible value of aaa
   Ans: Any of $\{\}$, $\{a\}$, $\{b\}$, $\{a, b\}$.

   (e) $\mathbb{P}(\mathbb{P}(X))$, $\mathsf{card}(\mathbb{P}(\mathbb{P}(X)))$.
   Ans: $\{\{\}, \{\{\}\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \ldots, \{\{\}, \{a\}, \{b\}, \{a, b\}\}\}$
   (The set of things taken none at a time (empty set, shown), one at a time (set containing the empty set (shown), then a set of each of the remaining elements of part b one at a time (shown), then sets of the elements of part b taken two at a time (not shown), three at a time (not shown), and finally four at a time (shown).)
   Ans: 16

   (f) $\mathbb{P}(\mathbb{P}(\mathbb{P}(X)))$, $\mathsf{card}(\mathbb{P}(\mathbb{P}(\mathbb{P}(X))))$.
   Ans: $\{\{\}, \{\{\}\}, \{\{\{\}\}\}, \{\{\{a\}\}\}, \ldots,$
   $\{\{\}, \{\{\}\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \ldots, \{\{\}, \{a\}, \{b\}, \{a, b\}\}\}$

   Ans: 65536 (2^{16})

   (g) $X \times \{0, 1\}$, $X \times \{\}$
   Ans: $\{(a, 0), (a, 1), (b, 0), (b, 1)\}$

   Ans: $\{\}$. (Note that $\mathsf{card}(A \times B) = \mathsf{card}(A) \times \mathsf{card}(B)$)

2. Given $X = \{a, b\}$ and $Y = \{0, 1\}$, show as sets of maplets:

   (a) $X \nrightarrow Y$
   Ans: various answers are possible, such as $\{(a, 0)\}$ (ordered pair notation) or $\{a \mapsto 1, b \mapsto 0\}$ (maplet notation) or $\{b \mapsto 1\}$

   (b) $X \rightarrow Y$
   Ans: $\{f \mid f \in X \nrightarrow Y \land \mathsf{dom}(f) = X\}$
   example: $\{a \mapsto 0, b \mapsto 1\}$

   What are the values of the following predicates?

   (c) $a \mapsto 0 \in X \nrightarrow Y$
   Ans: False (LHS should be a set)

   (d) $\{\{a \mapsto 0, a \mapsto 1\}\} \subseteq X \rightarrow Y$
   Ans: False (domain contains non-unique elements)

3. Students pass a subject if they gain at least 50 marks in the final examination. Given a function *results* : $\mathsf{STUDENTS} \nrightarrow \mathbb{N}$, that yields the examination result for a particular student, specify
   (a) the set of students that pass;
   Ans: $\{s \mid s \in STUDENTS \land s \in \mathsf{dom}(results) \land results(s) \geq 50\}$
   (Note: the $s \in STUDENTS$ clause can be omitted, since it is implied by $s \in \mathsf{dom}(results)$)
   (b) the set of students that fail.
   Ans: $\{s \mid s \in STUDENTS \land (s \notin \mathsf{dom}(results) \lor results(s) < 50)\}$
   (Note: the $s \in STUDENTS$ clause cannot be omitted, since s is otherwise unconstrained (untyped).

4. Memory on a modern computer can be considered to be an array of bytes.
   (a) define sets to represent memory.
   Ans: $BYTE = 0 \mathinner{..} 255$ (8-bit natural numbers)
   $MEMORY = 0 \mathinner{..} (n-1) \rightarrow BYTE$ where n is the size of the memory in bytes
   (b) show a function application to represent memory lookup of location *loc*.
   Ans: $MEMORY(loc)$
   (c) show a function update (relational override) to represent assignment of value *val* to memory location *loc*
   Ans: $MEMORY \lhd loc \mapsto val$

5. Assuming the following C declarations

```
int a[4] = {5,2,8,9};
char b[] = "the";
struct {int ca; char cb;} c;
```

give a set representation of the data structures `a`, `b`, `c`

Answer: All of these can be considered to be functions from VAR to $\mathbb{X}$, where VAR is the set of all variables, and $\mathbb{X}$ is the set of all result types (sets). Call this the set of variables $VARIABLES \in VAR \nrightarrow \mathbb{X}$

Answer: `a`: an example of a is $\{0 \mapsto 5, 1 \mapsto 2, 2 \mapsto 8, 3 \mapsto 9\}$
We have that $(a, x) \in VARIABLES \Rightarrow (a \in VAR) \land (x \in 0 \mathinner{..} 3 \rightarrow \mathbb{N}) \land (0 \mathinner{..} 3 \rightarrow \mathbb{N} \in \mathbb{X})$

Answer: `b`: an example of b is $\{0 \mapsto \texttt{'t'}, 1 \mapsto \texttt{'h'}, 2 \mapsto \texttt{'e'}\}$ (Note: the values `'t'`, `'h'`, `'e'` are assumed to stand for themselves (like the natural numbers)
We have that $(b, y) \in VARIABLES \Rightarrow (b \in VAR) \land (y \in 0 \mathinner{..} 2 \rightarrow \mathbb{C}) \land (0 \mathinner{..} 2 \rightarrow \mathbb{C} \in \mathbb{X})$. $\mathbb{C}$ is the set of all characters.

Answer: `c`: Structures are *variable composers*. They make variables out of composite names. There are two ways of viewing this. One is to think of ca, cb as functions that take an argument of type VAR, and then `c.ca` is really $ca(c)$. Alternatively, you can think of c as a function from $FIELDS$ to a new variable, and then regard `c.ca` as the function application $c(ca)$

First approach: an example of ca is $\{(c, 23)\}$ and cb is $\{(c, \texttt{'k'})\}$
We have that $\{(ca, x), (cb, y)\} \subset VARIABLES \Rightarrow (((ca \in VAR) \wedge (x \in VAR \nrightarrow \mathbb{X})) \Rightarrow (\exists\, p, q.(p, q) \in x \wedge q \in VAR \nrightarrow \mathbb{N} \wedge c \in \mathsf{dom}(q) \wedge VAR \nrightarrow \mathbb{N} \in \mathbb{X})) \wedge$ *similar expression for cb*

Second approach: an example of c is $\{(ca, 23), (cb, \texttt{'k'})\}$
We have that $(c, x) \in VARIABLES \Rightarrow (((c \in VAR) \wedge (x \in FIELDS \nrightarrow \mathbb{X})) \Rightarrow (\exists\, p, q.(p, q) \in x \Rightarrow (p = ca \Rightarrow q \in \mathbb{N} \wedge \mathbb{N} \in \mathbb{X}) \vee (p = cb \Rightarrow q \in \mathbb{C} \wedge \mathbb{C} \in \mathbb{X})))$

Other approaches are just as valid!

6. If we were modelling a taxi fleet company we might have three variables, *drivers*, *taxis* and *assigned* constrained by

$$\begin{aligned} drivers &: \mathbb{P}\,\mathsf{DRIVERS} \\ taxis &: \mathbb{P}\,\mathsf{TAXIS} \\ assigned &: drivers \rightarrowtail\!\!\!\!\!+ \; taxis \end{aligned}$$

where *drivers* is the set of drivers working for the company, *taxis* is the set of taxis owned by the company, and *assigned* is a function recording the assignment of drivers to taxis.

The arrow $\rightarrowtail\!\!\!\!\!+$ denotes a *partial injective* function. An injective function is a one-to-one function.

(a) Why is *assigned* a partial function?
Ans: Because not every driver need be assigned to a taxi.

(b) Why is *assigned* an injective function?
Ans: Because each taxi can be assigned to only one driver.

(c) Specify the drivers who are currently assigned.
Ans: $\mathsf{dom}(assigned)$

(d) Specify the drivers who are currently unassigned.
Ans: $drivers - \mathsf{dom}(assigned)$

(e) Specify the taxis that are currently assigned.
Ans: $\mathsf{ran}(assigned)$

(f) Specify the taxis that are currently unassigned.
Ans: $taxis - \mathsf{ran}(assigned)$

7. In the **SimpleLibrary** machine, we have three variables

$$\begin{aligned} books\_in\_library &\subseteq BOOK \\ books\_on\_shelf &\subseteq books\_in\_library \\ books\_on\_loan &\in books\_in\_library \nrightarrow users \end{aligned}$$

We want *books_on_shelf* and $\mathsf{dom}(books\_on\_loan)$ to *partition* the set *books_in_library*. Normally, we would specify this by:

$$\mathsf{dom}(books\_on\_loan) \cup books\_on\_shelf = books\_in\_library \wedge$$
$$\mathsf{dom}(books\_on\_loan) \cap books\_on\_shelf = \{\}$$

but it was specified by

$$\mathsf{dom}(books\_on\_loan) = books\_in\_library - books\_on\_shelf$$

(a) Show that the former follows from the latter.

Answer: By forming the union with *books_on_shelf* on both sides of the above equation, we get:

$$\mathsf{dom}(books\_on\_loan) \cup books\_on\_shelf = (books\_in\_library - books\_on\_shelf) \cup books\_on\_shelf$$

Since $(A - B) \cup B = A$, QED.

(b) Would $\mathsf{dom}(books\_on\_loan) \cup books\_on\_shelf = books\_in\_library$ have been equally effective.

Ans: No, because it doesn't require that a book on loan cannot be on a shelf as well.

8. For the Borrow(user,book) operation of the **SimpleLibrary** machine we get a proof obligation to show that

$$\mathsf{dom}(books\_on\_loan \cup \{book \mapsto user\}) = books\_in\_library - (books\_on\_shelf - \{book\})$$

Reason why this is true.

Answer: We have, from 7a,

$$\mathsf{dom}(books\_on\_loan) = books\_in\_library - books\_on\_shelf$$

So

$$\begin{aligned}\mathsf{dom}(books\_on\_loan \cup \{book \mapsto user\}) &= \mathsf{dom}(books\_on\_loan) \cup \{book\} \\ &= (books\_in\_library - books\_on\_shelf) \cup \{book\}\end{aligned}$$

But $book \in books\_on\_shelf$, so:

$$= \quad (books\_in\_library - (books\_on\_shelf - \{book\}) \cup \{book\}$$

But now, since $book \in books\_in\_library$, the trailing union can be elided (we know it is in *books_in_library*, and not in the thing being subtracted, so it remains in the final set:

$$= \quad books\_in\_library - (books\_on\_shelf - \{book\})$$

QED