# Math for CS 2019 Problem Set 2 solutions

https://github.com/spamegg1

May 21, 2022

## Contents

## 1 Problem 1.

### 1.1 (a)

Explain how to write a formula $\text{Members}(p, a, b)$ of set theory that means $p = \{a, b\}$.

*Proof.* Using the Hint, and using the $=$ defined in the problem statement, we have:

$\text{Members}(p, a, b) ::= \forall z.(z \in p \text{ IFF } (z = a \text{ OR } z = b))$ □

### 1.2 (b)

Explain how to write a formula $\text{Pair}(p, a, b)$, of set theory that means $p = \text{pair}(a, b)$.

*Proof.* We need to say that:

$z$ is an element of the pair $p$ iff either $z = a$ or $z = \{a, b\}$.

Using the Hint, and using Members(...) defined in part (a), we have:

Pair$(p, a, b)$ ::= $\forall z.(z \in p$ IFF $(z = a$ OR Members$(z, a, b)))$ □

### 1.3 (c)

Explain how to write a formula Second$(p, b)$, of set theory that means $p$ is a pair whose second item is $b$.

*Proof.* We need to say that there exists some set $a$ such that $p$ is Pair$(p, a, b)$, where Pair is defined in part (c). So we have:

Second$(p, b)$ ::= $\exists z.Pair(p, z, b)$ □

## 2 Problem 2

Prove De Morgan's Law for set equality

$$\overline{A \cap B} = \overline{A} \cup \overline{B}$$

by showing with a chain of IFF's that $x \in$ the left hand side iff $x \in$ the right hand side.

You may assume the propositional version of De Morgan's Law: NOT($P$ AND $Q$) is equivalent to $\overline{P}$ OR $\overline{Q}$.

*Proof.* Remember that the overline (or overbar) notation means "set complement". So $\overline{S}$ is the set of all elements (in the domain of discourse $D$) that are not in $S$.

$$\begin{array}{rccc}
x \in \overline{A \cap B} & \text{iff} & x \notin A \cap B & \text{(by definition of complement)} \\
 & \text{iff} & \text{NOT}(x \in A \cap B) & \\
 & \text{iff} & \text{NOT}(x \in A \text{ AND } x \in B) & \text{(by definition of intersection)} \\
 & \text{iff} & \text{NOT}(x \in A) \text{ OR NOT}(x \in B) & \text{(using De Morgan)} \\
 & \text{iff} & x \in \overline{A} \text{ OR } x \in \overline{B} & \text{(by definition of complement)} \\
 & \text{iff} & x \in \overline{A} \cup \overline{B} & \text{(by definition of union)}
\end{array}$$

□

## 3 Problem 3

### 3.1 (a)

Show that if $R$ and $S$ are c-d (concatenation definable), then so is $R \cap S$.

*Proof.* Let's carefully read through the definition of c-d. Remember "language" simply means "set of words", and $B$ is the set of all binary words:

A set of words is called concatenation-definable (c-d) if it can be constructed by starting with finite sets of words and then applying the operations of concatenation, union, and complement (relative to $B$) to these sets of words a finite number of times.

1. Assume $R$ and $S$ are c-d.

2. By the definition of c-d, there exist finite sets of words $R_1, \dots, R_m$ and $S_1, \dots, S_n$ such that,

$R$ is the result of applying concatenation, union, and complement to $R_1, \dots, R_m$ a finite number of times, and

$S$ is the result of applying concatenation, union, and complement to $S_1, \dots, S_n$ a finite number of times.

3. Notice that $R \cap S = \overline{\overline{R} \cup \overline{S}}$. (We can see this by applying complement to both sides of the equation from Problem 2.)

4. So $R \cap S$ can be obtained from $R$ and $S$ by applying complement 3 times and union once.

5. So $R \cap S$ can be obtained from the finite sets of words $R_1, \dots, R_m, S_1, \dots, S_n$ by applying concatenation, union and complement a finite number of times.

6. Therefore, $R \cap S$ is also c-d. □

## 3.2 (b)

Show that the language (set of words) consisting of the binary words that start with 0 and end with 1 is c-d.

*Proof.* Let's call this language (set of words) $X$.

The discussion below part (a) in the assignment proved that $B$ (the set of all binary words) is c-d.

Then we can obtain $X$ from $B$ by applying two concatenation operations:

$$X = \{0\} \cdot B \cdot \{1\} = \{abc \mid a \in \{0\} \text{ AND } b \in B \text{ AND } c \in \{1\}\}$$

This is the set of all words that start with 0, followed by any possible binary word, followed by 1. So that's exactly what we want.

Therefore $X$ is also c-d. □

## 3.3 (c)

Show that $0^*$ is c-d.

*Proof.* Let's read the definition of $0^*$.

First, 0 is just an abbreviation for the single element set $\{0\}$.

Second, the $S^*$ operation gives us the set of words you can get by concatenating any number of copies of words in S (including the empty word $\lambda$).

So we can write down $0^*$:

$$0^* = \{\lambda, 0, 00, 000, 0000, 00000, \ldots\}$$

How can we show that this set is c-d? Let's think about its complement, call it $X$. That would be the set of all binary words that contain at least one 1 in them. Is $X$ c-d?

We can write $X$ as: $X = B \cdot \{1\} \cdot B$. This is the set of all binary words of the form: start with any possible binary word (including the empty word), followed by a 1, followed by any possible binary word (including the empty word). That is exactly the set we are looking for (the set of all binary words that contain at least one 1).

Since $B$ is c-d, and $\{1\}$ is c-d (because it's finite), and we used only two more concatenation operations, $X$ is c-d. Therefore $0^* = \overline{X}$ is also c-d. □

## 3.4 (d)

Show that $\{01\}^*$ is c-d.

*Proof.* Again, just to get a better feel for it, let's write down $\{01\}^*$:

$$\{01\}^* = \{\lambda, 01, 0101, 010101, 01010101, 0101010101, \ldots\}$$

??? □

## 3.5 (e)

Explain why $\{00\}^*$ is not boring.

*Proof.* Let's read the definition of boring: $S$ is boring when either $S$ or $\overline{S}$ is 0-finite.

Let's read the definition of 0-finite: $S$ is 0-finite when $S \cap 0^*$ is a finite set of words.

So, rewriting the definition of boring:

$S$ is boring when either $S \cap 0^*$ or $\overline{S} \cap 0^*$ is a finite set of words.

Now let's write what it means to be "not boring":

$S$ is not boring when both $S \cap 0^*$ and $\overline{S} \cap 0^*$ are infinite sets of words.

With that out of the way, let's write down $\{00\}^*$:

$$\{00\}^* = \{\lambda, 00, 0000, 000000, 00000000, \ldots\}$$

We need to prove that both $\{00\}^* \cap 0^*$ and $\overline{\{00\}^*} \cap 0^*$ are infinite sets of words.

Since

$$0^* = \{\lambda, 0, 00, 000, 0000, \ldots\}$$

we can see that $\{00\}^*$ is actually a subset of $0^*$. So $\{00\}^* \cap 0^* = \{00\}^*$.

Since $\{00\}^*$ itself is infinite, so is $\{00\}^* \cap 0^*$.

What about $\overline{\{00\}^*} \cap 0^*$? Notice that

$$\overline{\{00\}^*} \cap 0^* = \{0, 000, 00000, \ldots\}$$

is the set of all words of odd number of 0s, which is also infinite. □

## 3.6 (f)

Verify that if $R$ and $S$ are boring, then so is $R \cup S$.

*Proof.* 1. Assume $R$ and $S$ are boring.

2. By definition of boring, either $R \cap 0^*$ or $\overline{R} \cap 0^*$ is finite, and either $S \cap 0^*$ or $\overline{S} \cap 0^*$ is finite.

3. Notice that, by distributive laws

$$(R \cup S) \cap 0^* = (R \cap 0^*) \cup (S \cap 0^*)$$

and, by one of De Morgan's laws (similar to Problem 2),

$$\overline{(R \cup S)} \cap 0^* = (\overline{R} \cap \overline{S}) \cap 0^* = (\overline{R} \cap 0^*) \cap (\overline{S} \cap 0^*)$$

4. From (2) there are 9 possible cases, but let's not go through them one by one. We can split them into two categories that are together exhaustive:

Case 1: At least one of $\overline{R} \cap 0^*$ and $\overline{S} \cap 0^*$ is finite.

Case 2: Both $\overline{R} \cap 0^*$ and $\overline{S} \cap 0^*$ are infinite (the negation of Case 1).

5. Consider Case 1. Then $(\overline{R} \cap 0^*) \cap (\overline{S} \cap 0^*)$ is finite. Therefore, by (3) $\overline{(R \cup S)} \cap 0^*$ is finite, so $R \cup S$ is boring.

6. Consider Case 2. Then by (2) both $R \cap 0^*$ and $S \cap 0^*$ are finite. Therefore $(R \cap 0^*) \cup (S \cap 0^*)$ is finite. Hence by (3) again $(R \cup S) \cap 0^*$ is finite, so $R \cup S$ is boring. □

## 3.7 (g)

Verify that if $R$ and $S$ are boring, then so is $R \cdot S$.

*Proof.* 1. Assume $R$ and $S$ are boring.

2. By definition of boring, either $R \cap 0^*$ or $\overline{R} \cap 0^*$ is finite, and either $S \cap 0^*$ or $\overline{S} \cap 0^*$ is finite.

3. Let's follow the Hint. Consider:

Case 1: Both $R \cap 0^*$ and $S \cap 0^*$ are finite.

Case 2: $R$ contains no all-0 words, including the empty word (i.e. $R \cap 0^* = \emptyset$).

Case 3: $S$ contains no all-0 words, including the empty word (i.e. $S \cap 0^* = \emptyset$).

Case 4: Not Cases 1, 2, 3. So: one of (or both) $R \cap 0^*$ and $S \cap 0^*$ is infinite; and both $R \cap 0^*$ and $S \cap 0^*$ are nonempty.

4. Consider Case 1. Let us think about the all-0 words of $R \cdot S = \{rs \,|\, r \in R, s \in S\}$.

If a word of the form $rs$ (where $r \in R$ and $s \in S$) is an all-0 word, then both $r$ and $s$ must also be all-0 words (possibly the empty word $\lambda$). Since both $R$ and $S$ contain only finitely many all-0 words, there are only finitely many all-0 words of the form $rs$.

Therefore $(R \cdot S) \cap 0^*$ is finite, so $R \cdot S$ is boring.

5. Consider Case 2. Assume $R$ contains no all-0 words (including the empty word). Then $R \cdot S = \{rs \,|\, r \in R, s \in S\}$ contains no all-0 words.

Why is this true? By a similar argument to the one in (4). If $rs$ is an all-0 word (where $r \in R$ and $s \in S$) then both $r$ and $s$ must be all-0 words (possibly the empty word). But $R$ contains no such word $r$ (not even the empty word $\lambda$).

Therefore $(R \cdot S) \cap 0^* = \emptyset$, so $R \cdot S$ is boring.

6. Consider Case 3. Similar to Case 2.

7. Consider Case 4. This is the difficult case. Assume that one of (or both) $R \cap 0^*$ and $S \cap 0^*$ is infinite; and both $R \cap 0^*$ and $S \cap 0^*$ are nonempty.

We want to prove that $\overline{(R \cdot S)} \cap 0^*$ is finite, which will prove that $R \cdot S$ is boring.

??? □

## 3.8 (h)

Explain why all c-d languages are boring.

*Proof.* All finite sets are boring: if $S$ is finite, then $S \cap 0^*$ is also finite.

By parts (f) and (g), unions and concatenations of boring sets are boring.

Also notice that if $S$ is boring, then $\overline{S}$ is also boring.

Therefore, all c-d languages are boring, because c-d languages are obtained from finite sets (which are all boring) by a finite number of union, concatenation, complement operations (which all preserve boringness by the above arguments). □