CS 103: Mathematical Foundations of Computing
Problem Set #8

Zara Rutherford, Diya Ahuja

March 8, 2024

Due Friday, March 8 at 4:00 pm Pacific

You'll submit your answers to Problem 1 and Problem 4 separately.
Symbols you might find helpful in this problem set:

- The "unstar" of a monoid from Q3 is written $M^\dagger$.

Problem Two: The Fixed-Point Theorem

i.

1. Yes, it is monotone
2. Yes, it is monotone
3. No, it is not monotone
4. Yes, it is monotone

ii.

1. Yes, aaa
2. Yes, aaa
3. Yes, a
4. Yes, $\varnothing^\star$

iii.

We need to prove that for a language L where $L \subseteq f(L)$, that $L \subseteq X$. To do so, pick an arbitrary $s \in L$. We will prove that $s \in X$. Since we know that $s \in L$, and $L \subseteq f(L)$, we see that $s \in X$ based on how we defined the language X.

iv.

We need to prove that if f is monotone, then $X = f(X)$. To do so, we will prove that $X \subseteq f(X)$ and $f(X) \subseteq X$.

First, we will show that $X \subseteq f(X)$. To do so, choose an arbitrary $s \in X$. We need to show that $s \in f(X)$. Since $s \in X$, by the definition of X we know that there exists a language L such that $s \in L$ and $L \subseteq f(L)$, so $s \in f(L)$. Based on our previous proof, we know $L \subseteq X$, and since f is monotone, this means $f(L) \subseteq f(X)$. Since we know $s \in f(L)$, this means $s \in f(X)$, as needed.

Second, we will show that $f(X) \subseteq X$. Since f is monotone, we know that $f(X) \subseteq f(f(X))$. Based on our previous proof, we see that if $f(X) \subseteq f(f(X))$, then $f(X) \subseteq X$, as needed.

Since $X \subseteq f(X)$ and $f(X) \subseteq X$, this means that $X = f(X)$, as required. $\blacksquare$

Problem Three: Unstarring a Language

i.

1. No, it is not a codeword of L
2. No, it is not a codeword of L
3. Yes, it is a codeword of L
4. No, it is not a codeword of L
5. No, it is not a codeword of L

ii.

1. $\{ \}$
2. $\{a, b\}$
3. $\{aa\}$
4. $\{b, ba, baa\}$

iii.

We need to prove that if M is a monoid over an alphabet Σ, then $(M^\dagger)^\star = M$. To do so, we will prove that $(M^\dagger)^\star \subseteq M$ and $M \subseteq (M^\dagger)^\star$.

First, we will show that $(M^\dagger)^\star \subseteq M$. To do so, we will first prove that $M^\dagger \subseteq M$. Choose an arbitrary $s \in M^\dagger$. We will show that $s \in M$. Since $s \in M^\dagger$, this means that s is a codeword of M. By definition of a codeword, we know that $s \in M$, as needed. Then, since $M^\dagger \subseteq M$, based on our proof in Problem Set Six we know that $(M^\dagger)^\star \subseteq M$, as required.

Second, we will show that $M \subseteq (M^\dagger)^\star$. Let $P(n)$ be the statement "for a string s with length n, if $s \in M$, then $s \in (M^\dagger)^\star$. We will prove that $P(n)$ holds for all $n \in \mathbb{N}$.

For our base case, we need to prove $P(0)$, that if the empty string is in M, then the empty string is also in $(M^\dagger)^\star$. By definition of a monoid, we know that $\varepsilon \in M$. By definition of the Kleene star, we know that $\varepsilon \in (M^\dagger)^\star$, as required.

For our inductive step, pick some $k \in N$ and assume $P(0), ..., P(k)$ are all true. We need to prove $P(k+1)$, that if a string s of length $k+1$ is in M, then $s \in (M^\dagger)^\star$. To do so, consider two cases:

Case 1: s is a codeword. This means that $s \in M^\dagger$, so by definition of the Kleene star, s must be in $(M^\dagger)^\star$, as needed.

Case 2: s is not a codeword. Since s is not a codeword, we know that there must exist two strings x and y in M such that $s = xy$, and neither x nor y are the empty string. Since s has length $k+1$ and both x and y have length at least equal to 1, this means that both x and y must have length between 1 and k, inclusive. Therefore, based on our inductive hypothesis, we know that $x \in (M^\dagger)^\star$ and $y \in (M^\dagger)^\star$. Furthermore, since $x \in (M^\dagger)^\star$ and $y \in (M^\dagger)^\star$, there re natural numbers m and n such that $x \in (M^\dagger)^n$ and $y \in (M^\dagger)^m$. This means that $xy \in (M^\dagger)^n(M^\dagger)^m = (M^\dagger)^{n+m}$, meaning that $s \in (M^\dagger)^{n+m}$. Thus, $s \in (M^\dagger)^\star$, as needed.

This shows that $P(k+1)$ holds, which completes the induction. ■

Problem Five: Executable Computability Theory

i.

- $L = \{ a^n b^n \mid n \in \mathbb{N} \}$
- Yes, the function is also a decider for L.
- Yes, L is decidable.

ii.

- $L = \{ w \mid w \in \Sigma^\star \}$
- Yes, the function is also a decider for L.
- Yes, L is decidable.

iii.

- $L = \{ w \mid w \in \Sigma^\star \text{and } |w| \text{ is in the fibonacci sequence} \}$
- No, the function is not a decider for L.
- Yes, L is decidable.

Problem Six: What Does it Mean to Solve a Problem?

i.

```
// By rejecting every input, if w ∈ Σ∗ or not, it is rejected so conditions 1 and 2 are satisfied
Start:
Return False
```

ii.

```
// All inputs are accepted
Start:
Return True
```

iii.

```
// Infinite loop and nothing is accepted or rejected
Start:
Goto Start
```

iv.

Consider a language L over an alphabet Σ for which there is a TM M that satisfies these three properties:
1. For all $w \in \Sigma^*$. M halts on w
2. For all $w \in \Sigma^*$, if M accepts w, then $w \in L$
3. For all $w \in \Sigma^*$, if M rejects w, then $w \notin L$

We will prove that L is a decidable language such that there is a decider for it. Since L has a TM M, we will show that M is a decider for L.

For M to be a decider, it must fulfill these properties:
1. For all $w \in \Sigma^*$. M halts on w
2. For all $w \in \Sigma^*$, $M \in L$ if and only if M accepts w.

Let's prove each property individually.

1. For all $w \in \Sigma^*$. M halts on w. We can see this is also the first property of M so that M fulfills this property of a decider.

2. For all $w \in \Sigma^*$, $M \in L$ if and only if M accepts w.

Let's first prove that for all $w \in \Sigma^*$, if $M \in L$ then M accepts w. We can prove this by contradiction such that there exists a $y \in L$ where M doesn't accept y.

Since M doesn't accept y and we know that M halts on y, it must be that M rejects y. However, the third property of M states that for all $w \in \Sigma^*$, if M rejects w, then $w \notin L$. Therefore, M must accept y so we have reached a contradiction and if $y \in L$ then M accepts y.

We also need to prove that for all $w \in \Sigma^*$, if M accepts w then $w \in L$. We can see that this is also one of the properties of M so we have proven this case.

We have proven each property such that M is a decider for L so that L is a decidable language. ■