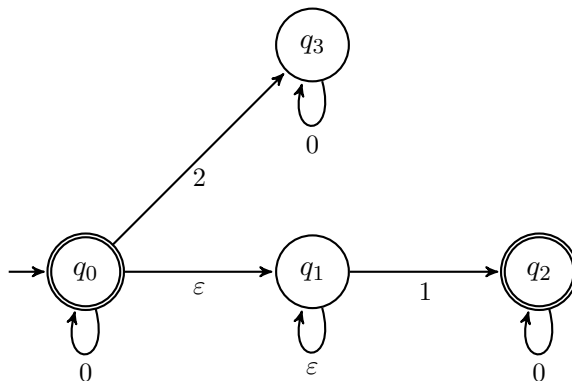# CSE 311: Foundations of Computing I
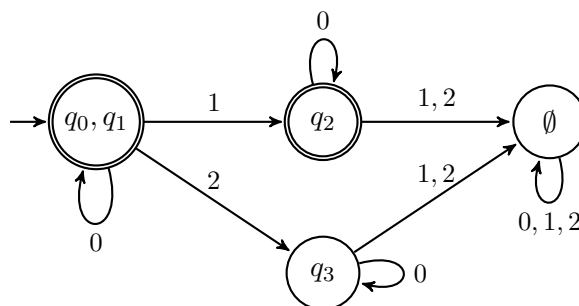
Section 10: Subset Construction, Irregularity, and Cardinality Solutions

1. Subset Construction
Convert the following NFA to a DFA:

Solution:

2. Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n 1^n 0^n \ : \ n \geq 0\}$ is not regular.

Solution:

Let $L = \{0^n 1^n 0^n \ : \ n \geq 0\}$. Let D be an arbitrary DFA, and suppose for contradiction that D accepts L. Consider $S = \{0^n 1^n : n \geq 0\}$. Since S contains infinitely many strings and D has a finite number of states, two strings in S must end up in the same state. Say these strings are $0^i 1^i$ and $0^j 1^j$ for some $i, j \geq 0$ such that $i \neq j$. Append the string 0^i to both of these strings. The two resulting strings are:

$a = 0^i 1^i 0^i$ Note that $a \in L$.

$b = 0^j 1^j 0^i$ Note that $b \notin L$, since $i \neq j$.

Since a and b end up in the same state, but $a \in L$ and $b \notin L$, that state must be both an accept and reject state, which is a contradiction. Since D was arbitrary, there is no DFA that recognizes L, so L is not regular.

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n (12)^m \ : \ n \geq m \geq 0\}$ is not regular.

Solution:

Let $L = \{0^n (12)^m \ : \ n \geq m \geq 0\}$. Let D be an arbitrary DFA, and suppose for contradiction that D accepts L. Consider $S = \{0^n : n \geq 0\}$. Since S contains infinitely many strings and D has a finite

number of states, two strings in S must end up in the same state. Say these strings are 0^i and 0^j for some $i, j \geq 0$ such that $i > j$. Append the string $(12)^i$ to both of these strings. The two resulting strings are:

$a = 0^i(12)^i$ Note that $a \in L$.
$b = 0^j(12)^i$ Note that $b \notin L$, since $i > j$.

Since a and b end up in the same state, but $a \in L$ and $b \notin L$, that state must be both an accept and reject state, which is a contradiction. Since D was arbitrary, there is no DFA that recognizes L, so L is not regular.

(c) Prove that the set of all Java programs is not regular.

Solution:

Let L = { public class { public static void main(String[] args) { $func(^i$ 4 $)^i$ }; public static int func(int i) { return 3; }} : $i \geq 1$} which is a subset of java programs we have chosen for the purposes of this proof.

Let D be an arbitrary DFA, and suppose for contradiction that D accepts L. Consider S={ public class { public static void main(String[] args) { $func(^i$: $i \geq 1$}. Since S contains infinitely many strings and D has a finite number of states, two strings in S must end up in the same state. Say these strings are:

S_a = public class { public static void main(String[] args) { $func(^a$
S_b = public class { public static void main(String[] args) { $func(^b$

Now we append both strings with $T_a = $ 4 $)^i$ }; public static int func(int i) { return 3; }} : $i \geq 1$} to get

$a = S_a T_a$
$b = S_b T_a$

Since a and b end up in the same state, but $a \in L$ and $b \notin L$, that state must be both an accept and reject state, which is a contradiction. Since D was arbitrary, there is no DFA that recognizes L, so L is not regular.

3. Cardinality

(a) You are a pirate. You begin in a square on a 2D grid which is infinite in all directions. In other words, wherever you are, you may move up, down, left, or right. Some single square on the infinite grid has treasure on it. Find a way to ensure you find the treasure in finitely many moves.

Solution:

Explore the square you are currently on. Explore the unexplored perimeter of the explored region until you find the treasure (your path will look a bit like a spiral).

(b) Here is a "proof" that the positive rationals, $\mathbb{Q}_+$, are uncountable.

Suppose for contradiction that the positive rationals are countable. Then there exists some listing of all elements $\mathbb{Q}_+ = \{q_1, q_2, q_3, \dots\}$. Note that each of these rationals q_i can also be written as an infinite decimal expansion. We define a new number $X \in \mathbb{Q}_+$ by flipping the diagonals of $\mathbb{Q}_+$; we set the ith digit of X to 7 if the ith digit of q_i is a 4, otherwise we set the digit to 4. This means that X differs from every q_i on the ith digit, so X cannot be one of q_i. Therefore our listing for $\mathbb{Q}_+$ was incomplete, which is a contradiction. Since the above proof works for any listing of the positive rationals $\mathbb{Q}_+$, *no* listing can be created for $\mathbb{Q}_+$, and therefore $\mathbb{Q}_+$ is uncountable.

What is the most significant error in this proof?

Solution:

X is not guaranteed to be a rational number (in fact, it almost certainly isn't), so X does not need to be in our listing of $\mathbb{Q}_+$ for our listing to be complete, so there is no contradiction.

(c) Prove that the set of irrational numbers is uncountable.
Hint: Use the fact that the rationals are countable and that the reals are uncountable.

Solution:

We first prove that the union of two countable sets is countable. Consider two arbitrary countable sets C_1 and C_2. We can enumerate $C_1 \cup C_2$ by mapping even natural numbers to C_1 and odd numbers to C_2.

Now, assume that the set of irrationals is countable. Then the reals would be countable, since the reals are the union of the irrationals (countable by assumption) and the rationals (countable). However, we have already shown that the reals are uncountable, which is a contradiction. Therefore, our assumption that the set of irrationals is countable is false, and the irrationals must be uncountable.

(d) Prove that $\mathcal{P}(\mathbb{N})$ is uncountable.

Solution:

Assume for the sake of contradiction that $\mathcal{P}(\mathbb{N})$ is countable. This means we can define an enumeration of elements $S_1, S_2, S_3, \ldots$ in $\mathcal{P}$.

Let s_i be the bit vector representation of S_i, i.e., where the n-th bit, denoted $s_i[n]$, is 1 if $n \in S_i$ and 0 otherwise. For example, for the set $0, 1, 2$, the bit vector would be $111000\ldots$

We then construct a new subset $X \subset \mathbb{N}$ such that $x[n] = \neg s_i[n]$ (that is, $x[n]$ is 1 if $s_i[n]$ is 0, and 0 otherwise). Note that X is not any of S_i since it differs from S_i on the n-th bit. However, X still represents a valid subset of the natural numbers, which means our enumeration is incomplete, which is a contradiction. Since the above proof works for any listing of $\mathcal{P}(\mathbb{N})$, *no* listing can be created for $\mathcal{P}(\mathbb{N})$, and therefore $\mathcal{P}(\mathbb{N})$ is uncountable.