Programming Languages and Computation

Week 12: Miscellaneous Problems

* 1. For each of the following statements about languages over the alphabet $\{0, 1\}^*$, determine if it is true or false.

(a) Every word of a regular language is of finite length.

(b) The language $\emptyset$ is regular.

(c) The language of all even length strings is regular.

(d) There is a language that is recognisable by an NFA but not recognisable by any DFA.

(e) Every finite subset of $\{0, 1\}^*$ is a regular language.

* 2. For each of the following statements, determine if it is true or false.

(a) In the diagram of a DFA, there is exactly one transition for each pair of state and letter.

(b) For each regular language S, there exists an NFA with a single accepting state that recognises S.

(c) The language Σ^* is regular.

(d) No DFA can accept the empty word.

(e) If an NFA with n states accepts any word, then it accepts a word of length $n-1$ or less.

* 3. For each of the following statements, determine if it is true or false.

(a) The following While program terminates when the value of `n` is initially the last digit of your student number.

```
while (5 <= n) {
  n := n + 1
}
```

(b) The following While program terminates when the value of `n` is initially the last digit of your student number and the value of `i` is initially `5`.

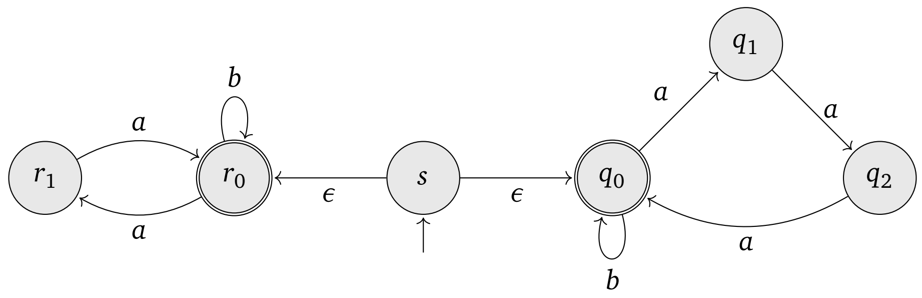

Figure 1: An Automaton.

```
while (i <= n) {
  i := i - 1
}
```

(c) The following While program terminates when the value of `n` is initially your student number and the value of `i` is initially negative.

```
while (i <= n) {
  i := i - 1
}
```

(d) `x := true` is a valid While program.

(e) Integer arithmetic in While can overflow.

* 4. For each of the following regular expressions over $\{a, b, c\}$, give (I) one word that is in the language denoted and (II) one word that is not.

Both words should be over the alphabet $\{a, b, c\}$. Label the two words with (I) and (II) so that it is clear which is claimed to be in and which is claimed to be not in.

(a) $(a + b)(a + b)(a + b)$

(b) $(a^*ba^*bc)^*$

(c) $(aa^* + bb^*)cc^*$

(d) $(\epsilon + b^*)^*$

* 5. Consider the automaton in Figure 1.

(a) Is this the diagram of a DFA? Justify your answer.

(b) Give two words that are accepted by the automaton.

(c) Give two words that are not accepted by the automaton.

(d) Describe in English the set of words accepted by the automaton (you may use any mathematical notation you find convenient).

* 6. Show that the function $f : \mathbb{N} \rightharpoonup \mathbb{N}$ defined by

$$f(x) \begin{cases} \simeq x+1 & \text{if } x^2 - 1 \text{ is at least } 2022 \\ \uparrow & \text{otherwise} \end{cases}$$

is computable.

* 7. State whether each of the following statements is true or false.

(a) Every injection has an inverse.

(b) The set $\mathbb{N}$ is decidable.

(c) Some WHILE programs compute total functions.

(d) If a function has an inverse, it must be a surjection.

(e) The set of all WHILE programs is countable.

* 8. Write a program that demonstrates that the function

$$f : \mathbb{N} \rightharpoonup \mathbb{N}$$
$$f(n) \begin{cases} \simeq 2^n & \text{if } n \text{ is divisible by } 3 \\ \uparrow & \text{otherwise} \end{cases}$$

is computable.

** 9. Construct a bijection $\mathbb{Z} \times \mathbb{Z} \xrightarrow{\cong} \mathbb{N}$. Prove that it is a bijection by constructing its inverse, and show that it is indeed an inverse.

** 10. For each of the following regular expressions, draw the diagram of an NFA that recognises the language denoted by the expression.

(a) $(01 + 001 + 010)^*$

(b) $0(10^*1 + 10)^*10$

(c) $(((00)^*(11))+01)^*$

*** 11. Suppose A is a regular language and $A \subseteq \{0,1\}^*$. Show that $A' = \{uv \mid u1v \in A\}$ is therefore also a regular language.

** 12. Write a While program, that, given 3 inputs given in variables `a`, `b`, and `c`, sorts them and returns the sorted values through the same variables. Your program can freely use other variables without needing to zero them out. (For example, running your program in memory $[\mathtt{a} = 42; \mathtt{b} = 12; \mathtt{c} = 1664]$ would yield a final memory that extends the mapping $[\mathtt{a} = 12; \mathtt{b} = 42; \mathtt{c} = 1664]$.)

*** 13. In this exercise, consider the following While program. This program always terminates (you can take this as given).

```
s := 0
while (0 <= n) {
  s := s + n
  n := n - 1
}
```

(a) Write down the final semantic configuration (in the While semantics) of the execution of the program when the state is initially such that the value of variable `n` is the final digit of your student number (and all other variables are initially set to 0, as usual.)

(b) Express the final value of variable `s` as an expression of the initial value of variable `n`.

(c) Identify a loop invariant that formally establishes that your previous answer is correct. You do not need to give a formal proof, but writing down your thoughts and reasoning may demonstrate understanding better than a partially correct answer.

** 14. Show that the predicate

$$U = \{\ulcorner S_1 \urcorner \mid \text{for all } k \leq 100 \text{ it is true that } [\![S_1]\!]_{\mathtt{x}}(k) + 1 = [\![S_2]\!]_{\mathtt{x}}(k+1)\}$$

is semi-decidable.

*** 15. Use the pumping lemma to show that the language $\{a^n b^m \mid n = 2 * m\}$ is not regular.

**** 16. Construct an NFA N with alphabet $\{1\}$ (consisting of a single letter: 1) and that satisfies the following two properties:

- N rejects at least one string.
- All strings of length less than or equal to the number of states of N are accepted by N.

** 17. Write a While program S_0 over a single variable r such that:

- The trace starting in configuration $\langle S, \sigma_0 \rangle$ with $\sigma_0 = \emptyset$ is infinite:

$$\langle S_0, \emptyset \rangle \to \langle S_1, \sigma_1 \rangle \to \langle S_2, \sigma_2 \rangle \to \langle S_3, \sigma_3 \rangle$$

- and every finite prefix of valuations of r:

$$\sigma_0(r), \sigma_1(r), \sigma_2(r), \ldots, \sigma_n(r)$$

(for any choice of n) is a word in the regular language $(0^+1^+2^+3^+)^*(0^+1^+2^+ + 0^+1^+ + 0^+ + \epsilon)$.

** 18. For each of the following languages over $\{a, b, c\}$, construct an NFA that recognises it. Your automata should have at most 4 states.

(a) $\{w \mid w$ contains substring “baa”$\}$

(b) $\{w \mid w$ ends with two occurrences of ’a’$\}$

(c) $\{w \mid w$ does not contain substring “abc”$\}$

** 19. Show that if the predicates $A \subseteq \mathbb{N}$ and $B \subseteq \mathbb{N}$ are decidable then their symmetric difference, i.e. the set

$$A \oplus B = \{x \in \mathbb{N} \mid (x \in A \wedge x \notin B) \vee (x \notin A \wedge x \in B)\} = (A \cup B) - (A \cap B)$$

is also decidable.

*** 20. Show that the following language over $\{0, 1\}$ is regular and justify your answer:

$$\{0^k u 0^k \mid k \geq 1 \wedge u \in \{0, 1\}^*\}$$

*** 21. Let Σ be some alphabet. We say that $a_1 a_2 \cdots a_m$, with each $a_i \in \Sigma$, is a subword of $w \in \Sigma^*$ just if there are words v_0, v_1, ...v_m over Σ such that $w = v_0 a_1 v_1 a_2 v_2 \cdots a_m v_m$.

For a given language A over Σ, define

$$\mathsf{SUBWORDS(A)} = \{u \mid \exists w \in A.\ u \text{ is a subword of } w\}$$

Prove that the class of regular languages is closed under the $\mathsf{SUBWORDS}$ operation.

*** 22. Show that the predicate

$$V = \{\phi(\gamma(S_1), \gamma(S_2)) \mid \text{there exists } k \in \mathbb{N} \text{ such that } [\![S_1]\!]_{\mathtt{x}}(k) + 1 = [\![S_2]\!]_{\mathtt{x}}(k+1)\}$$

is undecidable. (The use of $=$ above means that both sides of the equation must be defined, and equal.)

*** 23. Show that the predicate

$$U = \{\langle \ulcorner S \urcorner, \ulcorner T \urcorner \rangle \mid [\![S]\!]_{\mathtt{x}}(0) \simeq [\![T]\!]_{\mathtt{x}}(0)\}$$

is undecidable.

**** 24. One of the following two languages over the alphabet $\{0, 1, 2\}$ is regular and the other is not. Which is which? Give a full justification for your answer.

$$\{uv \mid \#_0(u) = \#_1(v)\}$$
$$\{u2v \mid \#_0(u) = \#_1(v)\}$$

Here, for any word w, $\#_0(w)$ is the number of occurrences of 0 in w and $\#_1(w)$ is the number of occurrences of 1 in w. For example, for $w = 0010101$, $\#_0(w) = 4$ and $\#_1(w) = 3$.