Review of [1]

200 Problems on Languages, Automata, & Computation
by **Filip Murlak, Damian Niwiński, Wojciech Rytter**

Cambridge University Press, 2023
Hardback, $90.00; Paperback, $39.99; eBook, $39.99; 254 pages

Review by **William Gasarch** (`gasarch@umd.edu`)

1 Introduction

The title of the book describes its content so well that I have little to add except this: Since the phrase *Languages, Automata, & Computation* may mean different things to different people, I list the chapters:

1. Part I: Problems
   (a) Words, Numbers, Graphs
   (b) Regular Languages
   (c) Context-free Languages item Theory of Computation (This chapter includes computational complexity.)
2. Part II: Solutions
   (a) Words, Numbers, Graphs
   (b) Regular Languages
   (c) Context-free Languages
   (d) Theory of Computation
3. Further Reading
4. Index

2 Summary

Each problem in this book has one of four markings (if you count NO marking as a marking).

1. Easy- a Five Pointed Star with white interior.

2. Intermediate-unmarked,

3. Hard-$\star$,

4. Very Hard-$\star\star$.

Solutions are included, which is either good or bad depending on how you intend to use it. Since some of the problems are hard, I like that there are solutions.

Are these good problems? Absolutely yes. There were many that I have not seen or thought of before, and note that I have taught automata theory $\aleph_0$ times and TAed it four times. There is a wide range of problems. The chapter headings help with finding problems in various topics. The $\star$-system helps with finding problems of various difficulties.

The rest of this review will be a few problems from each chapter and an opinion. When I give a problem, I paraphrase it since there are legal issues with quoting a book verbatim.

Elsewhere in this book review column is a review of *The Logical Approach to Automatic Sequences: Exploring Combinatorics on Words with Walnut* by Jeffrey Shallit. That book describes a program called *Walnut* which can be used to solve many problems having to do with sequences. Some of the problems in the book *200 Problems* can probably be solved automatically with *Walnut* (including some mentioned in this review). This is *not* a criticism; however, the user of *200 Problems* should be aware of this issue. There is a large discussion to be had about what-tool-do-we-allow-our-students-to-use, which we will not engage in here.

2.1 Words, Numbers, Graphs

I expected this chapter to have all easy and intermediate problems. It did not! It has two easy, two intermediate, two hard, and one very hard problems.

$\star\star$ **Definition** The *Thue-Morse sequence* is defined as follows: The first symbol is 0. Assume you have the first n symbols t_n. The next n symbols are t'_n, defined as t_n with the 0's changed to 1's and the 1's changed to 0's. (The problem gives two definitions and asks you to prove they are equivalent.)

1. Show that the Thue-Morse sequence is *cube-free*. That is, there is no subword of the form www where $w \in \{0,1\}^+$. (The Thue-Morse sequence is actually *strongly cube-free*: there is no subword of the form $bwbwb$ where $b \in \{0,1\}$ and $w \in \{0,1\}^+$.)

   They give a hint, but I will not. So perhaps my version is $\star\star\star$.

2. Construct a sequence over a 4-letter alphabet that is square-free. That is, there is no subword of the form ww where $w \in \{0,1\}^+$.

3. Construct a sequence over a 3-letter alphabet that is square-free.

4. Is there a sequence over a 2-letter alphabet that is square-free?

2.2 Regular Languages

1. Let $\Sigma = \{0, \ldots, 9\}$ and view the input as a number in base 10. Show that

$$\{w \colon w \equiv 0 \pmod 7\}$$

is regular.

2. $\star$ An *infix* of a word $w = \sigma_1 \cdots \sigma_n$ is any string of the form $\sigma_i \cdots \sigma_j$ where $i \leq j$. A language L is *closed under infix* if every infix of every word in L is also in L.

   Give an example of an infinite language that is closed under infix but does not contain an infinite regular language as a subset.

3. $\star\star$ Let $\Sigma = \{a, b\}$. If w is a word over Σ and $\sigma \in \{a, b\}$ then let $\#_\sigma(w)$ be the number of σ's in w.

   Give an algorithm for the following (it need not be efficient):

   (a) Given a DFA for L, determine whether for all $w \in L$, $\#_a(w) = \#_b(w)$.

   (b) Given a DFA for L, determine whether there exists an infinite number of $w \in L$ such that $\#_a(w) = \#_b(w)$.

2.3 Context-free Languages

1. Give an algorithm to determine whether $L(G)$ is infinite for a given CFG G.

2. $\star$ Show that for every context-free grammar G there is a constant C such that every non-empty word w generated by G has a derivation of length at most $C|w|$.

3. $\star$ Show that the set of palindromes cannot be recognized by a deterministic pushdown automaton.

2.4 Theory of Computation

1. $\star$ Let $X \subseteq \mathsf{N}$. Show that X is decidable iff either X is finite or X is the image of a computable strictly-increasing function. (I think this question should be 0 stars.)

2. $\star$ Is the following problem decidable: given $u, v \in \Sigma^*$ and a number k, is there a string w of length at least k such that $\#_u(w) = \#_v(w)$?

3. Show that the following problem is NP-complete: given a regular expression α over Σ ($|\Sigma|$ might be large), is there a word $w \in L(\alpha)$ such that every letter in Σ appears in it. (I think this question should be $\star$.)

3 Opinion

I began reading this book skeptical that I would find problems in it that I had not already seen. I was wrong. There are many great problems in this book, some for your students as homework, some to be the basis of lectures you will give your students (I will definitely teach the last problem on Theory of Computation that I stated the next time I teach Automata theory), and some for your own edification.

The only criticisms are that there is not enough problems on NP-completeness or complexity theory in general.