1. Consider the following random access Turing machine:

1. load =1 // Store 1 in R0
2. store 1 // Store contents of R0 in R1
3. load =0 // Store 0 in R0
4. store 2 // Store contents of R0 in R2
5. read 1 // Read T[R1] into R0
6. jzero 13 // If contents of R0 == 0, go to 13
7. add 2 // Add contents of R2 to R0
8. store 2 // Store contents of R0 in R2
9. load 1 // Load contents of R1 into R0
10. add =1 // Add 1 to contents of R0
11. store 1 // Store contents of R0 in R1
12. jump 5 // Go to 5
13. load =1 // Set R0 to 1
14. store 1 // Put the 1 into R1
15. load 2 // Load contents of R2 into R0
16. write 1 // Write R0 contents to T[R1] = T[1]
17. halt

(a) If the input tape contains the following pairs of decimal integers, find the output of the machine.

\[(1, 1)(2, 3)(3, 2)(4, 1)(5, 0)\]

(b) More generally, describe what the machine does if its input tape has the “proper” format?

2. First part of problem 4.5.2 on page 227.

3. Use the grammar in example 4.6.2 on page 229 to derive the string \textit{aabbcc}.
4. Problem 5.2.1, parts (a), (b), and (c) on page 250.

5. Consider the language

\[ H = \{ "M" \, w \, : \, \text{Turing machine } M \text{ halts on input } w \}. \]

In class we saw that \( H \) is recursively enumerable. We also saw that if \( H \) is recursive, then every recursively enumerable language is also recursive. This problem looks at the remainder of the proof that \( H \) is not recursive. (See page 252, paragraphs 2, 3, and 4.)

(a) Consider the language

\[ H_1 = \{ "M" \, : \, \text{Turing machine } M \text{ halts on input } M \}. \]

Show that if \( H \) is recursive, then \( H_1 \) is recursive.

So in order to show that \( H \) is not recursive, it suffices to show that \( H_1 \) is not recursive.

(b) If \( H_1 \) were recursive, then its complement

\[ \overline{H}_1 = \{ w \, : \, w \text{ is not the encoding of a Turing machine} \}
\cup \{ "M" \, : \, \text{Turing machine } M \text{ does not halt on input } M \} \]

would also be recursive. Why?

(c) If \( \overline{H}_1 \) isn’t recursively enumerable, explain why it can’t be recursive.

(d) Show that \( \overline{H}_1 \) isn’t recursively enumerable.