Relevant readings: Cutland Ch. 4, 5, §6.1

1. (Cutland 4.3.2, 4.3.3)
   (a) Let \( f_0, f_1, \ldots \) be an enumeration of (some) partial functions from \( \mathbb{N} \) to \( \mathbb{N} \). Construct a function \( g \) from \( \mathbb{N} \) to \( \mathbb{N} \) such that \( \text{Dom}(g) \neq \text{Dom}(f_i) \) for each \( i \).
   (b) Let \( f \) be a partial function from \( \mathbb{N} \) to \( \mathbb{N} \) and let \( m \in \mathbb{N} \) be fixed. Construct a non-computable function \( g \) such that
   \[
g(x) \approx f(x) \quad \text{for } x \leq m.
   \]

2. (Cutland 5.3.2)
   (a) Show that there is a total computable function \( k(e) \) such that for any \( e \), if \( \phi_e \) is the characteristic function for a decidable predicate \( M(x) \) then \( \phi_{k(e)} \) is the characteristic function for “not \( M(x) \)”.
   (b) Show that there is a total computable function \( s(x, y) \) such that for all \( x, y \)
   \[
   E_s(x, y) = E_x \cup E_y.
   \]

3. (Cutland 6.1.8.2) Show that the following problems are undecidable.
   (a) “\( W_x = W_y \)” Hint: reduce “\( \phi_x \) is total” to this problem
   (b) “\( W_x = \emptyset \)”.

The following two questions study a particular non-computable function: the “Busy Beaver”. Define

\[
B(n) = \text{maximum output of any URM program with at most } n \text{ many instructions, on input } 0.
\]

Note that \( B \) is a total function: for each \( n \), there are finitely many programs with at most \( n \) many instructions and at least one of these has some output.

4. Prove that if the halting problem were solvable then \( B \) would be computable.

5. However, the halting problem is not solvable so there remains some work to be done.
   (a) Prove that \( B \) is a strictly increasing function (for all \( n \), \( B(n) < B(n + 1) \)).
   (b) Prove that for all \( n \geq 1 \), \( B(n + 5) \geq 2n \).
   \[\text{Hint: how many instructions does it take to double the contents of a register?}\]
   (c) Recall the definition of one function dominating another from PS 2. Also, recall the result from that problem set: any URM computable function \( f \) is dominated by a strictly increasing URM computable function \( g \).
   – Note that if \( g(n) \) is defined but \( f(n) \) is not, we say that \( f(n) \leq g(n) \).
   Therefore, prove that for any URM computable function \( f \), \( B \) dominates \( f(n) + 1 \).
   \[\text{Hint: given an increasing URM computable function } g, \text{ find a program that witnesses that } B(n + k_0) > g(B(n)) \text{ for some number } k_0\]
   (d) Finally, prove that \( B \) is not computable.