CS 517: Problem Set 2

1. With M a Turing machine, define $T(M)$ to be the number of steps that M runs on empty input before halting. $T(M) = \infty$ if M doesn't halt. For every n there are only finitely many Turing machines with n states. Define $B(n)$ to be the largest **finite** $T(M)$ among n-state Turing machines M. In other words, $B(n)$ is the longest number of steps an n-state TM can run while still halting.

   We say that a function $f : \mathbb{N} \to \mathbb{N}$ is computable if there is a TM that on input n (encoded in binary), halts with $f(n)$ (in binary) written on its tape.

   Prove that if $f(n) \geq B(n)$ for all n, then f is **not** computable. (In other words, B grows faster than any computable function.)

   Hint: Use the idea of a reduction. But since we are talking about a function and not a decision problem, expect things to be slightly different.

2. Please recall the definition of a context-free grammar, and remember that sometimes a single string can have more than one valid parse tree. For example, the string *aaabbbbb* has several possible parse trees in the grammar $S \to aSbb \mid aSb \mid ab$.

   Show that the following problem is undecidable: Given a context-free grammar G, decide whether there exists any string with two parse trees in G.

   Hint: give a reduction from the PCP problem.

3. Show that the following language is undecidable.

   $\{\langle M, x, c\rangle \mid M \text{ is a TM that on input } x \text{ eventually writes character } c \text{ somewhere on its tape}\}$

   Note that this language is **not** covered by Rice's theorem because it is not about a property of $L(M)$. Rather, it is about some *internal* property of M's computation.