Throughout thisHW $M_1, M_2, \ldots$ is a standard list of Turing Machines. Can also view as a list of all partial computable functions.

1. (60 points — 15 points for each part)
   (a) Let $M$ be a Turing machine. Show that the following set is $\Sigma_1$:
   $$\{ x \mid M(x) \downarrow \}$$
   (b) Describe an algorithm $M$ such that
   $$\{ x \mid M(x) \downarrow \}$$
   is undecidable.
   (HINT- Write an $M$ such that the set
   $$\{ x \mid M(x) \downarrow \}$$
   is HALT. Recall that HALT is
   $$\{ e \mid M_e(e) \downarrow \}$$
   )
   (c) Let $M$ be a Turing machine. Show that the following set is $\Sigma_1$:
   $$\{ y \mid \text{there is some } x \text{ such that } M(x) = y \}$$
   (d) Describe an algorithm $M$ such that
   $$\{ y \mid \text{there is some } x \text{ such that } M(x) = y \}$$
   is undecidable.
   (HINT- Write an $M$ such that the set
   $$\{ y \mid \text{there is some } x \text{ such that } M(x) = y \}$$
   is HALT.)

1
2. (40 points — 20 points each) A NATHAN program is a program that can, on each input, make 10 queries to HALT.

(a) Is there a NATHAN program for the following problem: on input \((e_1, \ldots, e_{100})\) determine EXACTLY which \(e_i\) are such that \(M_{e_i}(0) \downarrow\)? (Formally the output is a bit string \((b_1, \ldots, b_{100})\) such that, for all \(1 \leq i \leq 100\),

\[
M_{e_i}(0) \downarrow \text{ iff } b_i = 0.
\]

(b) Is there a NATHAN program for the following problem: on input \(n\) viewed as a number written in binary, output some string \(y\) such that \(C(y) \geq n\) (\(C(y)\) is the Kolmogorov complexity of \(y\) — the size of the smallest Turing Machine that prints out \(y\) on input 0.)