Question 1. [35 points] The “Invitation” of the Preface of the Hemaspaandra-Ogihara textbook makes the surprising claim that every “student of computer science already possess the ability required to understand, enjoy, and employ complexity theory.” It makes this claim because the book’s view is that at the heart of complexity theory is: algorithms (that is the secret 1 version)/simple algorithms (that is the secret 2 version—in your answer, you did not have to say whether you answer was regarding secret 1 or was regarding secret 2) (fill this in, either in the version of secret 1 of the invitation or in the version of secret 2).

Question 2. [30 points] Give the definition of HP from the slides.

\[
HP = \{ x \mid M_x(x) \text{ accepts} \}
\]

Question 3. [35 points] Prove that if \( A \) is a coRE set and \( B \) is a coRE set and \( A \cap B = \emptyset \), then there exists a recursive set \( C \) such that \( A \subseteq C \subseteq B \).

(Hint: think of last class and the proof that sets that are RE and coRE are recursive, but now with a bit of a twist!) (It is fine if your answer continues on the back of this page.)

Since \( A \) and \( B \) are coRE, \( \overline{A} \) and \( \overline{B} \) are RE and thus TMs exist that accept both of these languages. We will construct a TM \( T \) which will accept such a set as follows.

\( T \) will simulate TMs for \( \overline{A} \) and \( \overline{B} \) on its input in a dovetailed fashion. If the \( \overline{A} \) TM accepts, then halt and reject. If the \( \overline{B} \) TM accepts, then halt and accept.

\( T \) will halt on every input, as \( \overline{A} \cup \overline{B} = \Sigma^* \). Thus the language of \( T \) is recursive. \( T \) will accept every element of \( A \), as each of these will eventually be accepted by the \( \overline{B} \) TM, since \( A \subseteq \overline{B} \). \( T \) will reject every element of \( B \), since each of these will be accepted by the \( \overline{A} \) TM, since \( B \subseteq \overline{A} \). Inputs which are in \( \overline{B} - A \) will either be accepted or rejected depending on which of the TMs finishes first. Regardless, \( A \subseteq L(T) \subseteq \overline{B} \). Therefore \( L(T) \) is such a set.