- Start as early as possible, and contact the instructor if you get stuck.
- See the course outline for details about the course's grading policy and rules on collaboration.
- Submit your completed solutions to **Crowdmark**.

1. **Descriptions of natural numbers [6 marks]**

   Definition 1. Given an integer $n \geq 0$, the *exponential-Golomb encoding* of n is

   $$\langle n \rangle := \mathsf{0}^{|x|}\mathsf{1}x^R$$

   where $x \in \{\mathsf{0}, \mathsf{1}\}$ is such that $\mathsf{1}x$ is the binary representation of $n+1$.

   There exists a Turing machine M which on input $\langle u \rangle \langle v \rangle$ for all integers $u, v \geq 0$, inserts a separator between the representations,

   $$\langle u \rangle \texttt{\#} \langle v \rangle,$$

   and then halts. Explain the low-level steps the Turing machine does to accomplish this. You may extend the tape alphabet in your Turing machine if it helps.

2. **Red-Blue Turing machines [12 marks]**
   For the 90^{th} anniversary of the invention of the Turing machine, they are introducing a special limited-edition *red-blue Turing machine* (RBTM). A red-blue Turing machine has the following additional rules:
   - Every state is coloured red or blue.
   - The initial state, q_0, is red and the unique final state, q_{finish}, is blue.
   - There are no transitions from a blue state to a red state.

   As a result, there are three outcomes for a RBTM: it loops forever on the red states (Red), it loops forever on the blue states (Blue), or it halts (Halt). Coincidentally, a standard Turing machine also has three outcomes: halt and accept (Acc), halt and reject (Rej), or loop forever (Loop).

   (a) Prove the following theorem (an RBTM can simulate a TM).

   Theorem 2. *There exists a bijection* $f\colon \{\mathsf{Red}, \mathsf{Blue}, \mathsf{Halt}\} \to \{\mathsf{Acc}, \mathsf{Rej}, \mathsf{Loop}\}$ *and a red-blue Turing machine* M' *with the property that on all inputs* $\langle M \rangle x$ *(where* M *is a Turing machine and* $x \in \{\mathsf{0}, \mathsf{1}\}^*$ *is a bitstring), the outcome of* M' *is* $o \in \{\mathsf{Red}, \mathsf{Blue}, \mathsf{Halt}\}$ *if and only if* M *does* $f(o)$ *on input* x*.*

   (b) Prove the following theorem (a TM *cannot* simulate an RBTM).

   Theorem 3. *For each bijection* $g\colon \{\mathsf{Acc}, \mathsf{Rej}, \mathsf{Loop}\} \to \{\mathsf{Red}, \mathsf{Blue}, \mathsf{Halt}\}$*, there does not exist a Turing machine* M *such that on input* $\langle M' \rangle x$ *(where* M' *is a RBTM and* $x \in \{\mathsf{0}, \mathsf{1}\}^*$*), the outcome is* $o \in \{\mathsf{Acc}, \mathsf{Rej}, \mathsf{Loop}\}$ *exactly when* M' *does* $g(o)$ *on input* x*.*

   (c) Discuss (two or three sentences) how this relates to the Church-Turing thesis.

3. **Logic and Computability [8 marks]**
   Fix a logical system, like Zermelo-Fraenkel set theory (ZF). The details of the theory don't matter as long as we have three facts.

- Theorems and proofs can be represented as bitstrings.
- There is a Turing machine ProofChecker which given a theorem and a proof will check if the proof correctly proves the theorem.
- The system is consistent – one cannot prove a contradiction.

Famously, it is not possible to prove the consistency of a theory like ZF within the theory itself, so we have to take the last point on faith.

Assuming the consistency of ZF, prove the following theorem.

Theorem 4. *There exists a TM which loops, but there is no* ZF *proof it loops.*

Hint: You can run more than just two computations in parallel.

4. **States vs. Description Length [10 marks]**
Turing machines are unusual because we often measure a TM's size by the *number of states*[1] but this does not coincide with the length of the description, not even up to constant factors (i.e., big-Θ).
For this question, suppose the tape alphabet $\Gamma = \{\square, \mathtt{0}, \mathtt{1}\}$ is fixed, along with $b = \square$ and $\Sigma = \{\mathtt{0}, \mathtt{1}\}$. Write the states $Q = \{0, 1, \ldots, n, n+1\}$ as integers, with $q_0 = 0$ being the first state and $F = \{q_{accept}, q_{reject}\} = \{n, n+1\}$ being the last two.
   (a) Given n, the only part of the Turing machine left to specify is $\delta \colon (Q \backslash F) \times \Gamma \to Q \times \Gamma \times \{L, R\}$, the transition function. Argue that there is a description $\langle \delta \rangle \in \{\mathtt{0}, \mathtt{1}\}^*$ for δ (and thus the TM) of length $O(n \log n)$.
   (b) On the other hand, prove that the description must be at least $\Omega(n \log n)$ bits long because there are $2^{\Omega}(n \log n)$ *semantically* different Turing machines. That is, there is a set of $2^{\Omega(n \log n)}$ Turing machines and for every pair there exists an input where they differ (one accepts and the other rejects, or loops, etc.)
   Hint: DFAs are simple Turing machines that never change the tape symbol and always move left.
   (c) In the lecture on the recursion theorem, we saw that for any bitstring $s \in \{0, 1\}^*$ there is a *printer* P_s which prepends s to the input string. We also saw FindPrinter, which replaced input s with $\langle P_s \rangle$. The printer used $|s| + 3$ states, but we now know that $\langle P_s \rangle$ is asymptotically longer than the string P_s prints.
   Prove that there exists a family of printers $\{P'_s\}_{s \in \{\mathtt{0},\mathtt{1}\}^*}$, and a FindPrinter′ which generates $\langle P'_s \rangle$ given s, where the length of $\langle P'_s \rangle$ is $O(|s|)$.

[1] Busy beavers, for example, are classified by number of states.