C1 Theory (25 points)

Let N denote the set of non-negative integers.

Let $\Sigma = \{a, b\}$.

In this problem (C1) we will make reference to the languages L_1, L_2, and L_3 defined just below.

Definition

(i) $L_1 \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid \textit{in at least one prefix of } w \textit{ there are exactly 4 more } b\textit{'s than } a\textit{'s}\}$.[1]

(ii) $L_2 \stackrel{\text{def}}{=} \{a^i b^j \mid i, j \in N \ \wedge \ i \neq j\}$.

(iii) $L_3 \stackrel{\text{def}}{=}$ *the set of all* $w \in \Sigma^*$ *such that every other symbol of* w *is an* a*, but starting at the first symbol if* $|w|$ *is odd and starting at the second symbol for* $|w|$ *even.*[2]

a. (8 points)
Use Myhill-Nerode to show that L_1 is *not* regular.[3]

b. (8 points)
Use Pumping *directly* to show that L_2 is not regular. *No fair* working with $\overline{L_2}$ or some such.

Hint for C1b: Consider the word $a^n b^{n+n!}$ for *suitable* n.

c. (9 points)
Prove that L_3 *is* regular. N.B. If you do this by programming in *relevant* system(s), also *document* your "code". *Your documentation should be to help the graders understand what you had in mind.*

---

[1] For example, $aaaabbbaabbbbbbbabba \in L_1$ since the prefix $aaaabbbaabbbbbbb$ has 10 b's and 6 a's.

[2] For example, $0, abaaa, baba \in L_3$, where '0' denotes the empty word. 0 is in L_3 since it contains no counterexamples.

[3] Recall that the crucial equivalence relation for Myhill-Nerode applied to the language L_1 is $\equiv_{L_1}$, where, for words $x, y \in \Sigma^*$,

$$x \equiv_{L_1} y \stackrel{\text{def}}{\Leftrightarrow} (\forall z \in \Sigma^*)[xz \in L_1 \Leftrightarrow yz \in L_1]. \tag{1}$$

C2 **Theory** (25 points)

Suppose n is a *fixed* integer > 1. Let $\Sigma = \{a, b\}$. Let

$$L = \{a,b\}^* \cdot \{a\} \cdot \{a,b\}^{n-1} \tag{2}$$

a. (10 points)
Explicitly exhibit the state diagram of a non-deterministic finite automaton, $\mathcal{M}_1$, having *exactly* $n+1$ states and such that $L(\mathcal{M}_1) = L$, where L is defined in (2) above.

b. (15 points)
For this problem (C2b) you may use Myhill-Nerode.[4]
Suppose $\mathcal{M}$ is *any* (deterministic) finite automaton such that $L(\mathcal{M}) = L$ (again, where L is defined in (2) above). *Prove* that $\mathcal{M}$ has *at least* 2^n *states*.

Hint for C2b: Suppose $x, y \in \Sigma^n$ and $x \neq y$.
Consider an i such that

$$[1 \leq i \leq n \ \wedge \ [x, y \text{ differ in their } i\text{-th symbol (reading left to right)}]]. \tag{4}$$

For *each* of the words xb^{i-1} and yb^{i-1} carefully show exactly how many symbols there are from the $i+1$-st *through* the last, *inclusive*.
Use the above to show that

$$[xb^{i-1} \in L \Leftrightarrow yb^{i-1} \notin L]. \tag{5}$$

Conclude from (5) just above something interesting and relevant about xb^{i-1}, yb^{i-1}, and $\equiv_L$.
Develop *and justify* a useful formula for $|\Sigma^n|$.
Explain *clearly* how showing the above is *relevant* to solving C2b.

[4] Recall that the crucial equivalence relation for Myhill-Nerode applied to L from (2) above is $\equiv_L$, where, for words $x, y \in \Sigma^*$,

$$x \equiv_L y \overset{\text{def}}{\Leftrightarrow} (\forall z \in \Sigma^*)[xz \in L \Leftrightarrow yz \in L]. \tag{3}$$

C3 Theory (25 points)

Let N denote the set of non-negative integers.

We write real numbers $\mathbf{r}$ in the closed interval between the real numbers $\mathbf{0}$ and $\mathbf{1}$ (inclusive) in binary notation as

$$.r_0r_1r_2r_3\cdots, \tag{6}$$

where each r_i is a 0 or a 1 (a bit) and where

$$\mathbf{r} = \sum_{i \in N} \frac{r_i}{2^{i+1}}.^5 \tag{7}$$

Definition *Suppose* $\mathbf{r}$ *is a real number in the closed interval between the real numbers* $\mathbf{0}$ *and* $\mathbf{1}$ *(inclusive), where* $\mathbf{r}$ *is written in binary as in (6) above.*

(ii) *We say* $\mathbf{r}$ *is* computable $\stackrel{\text{def}}{\Leftrightarrow}$ *the function*

$$\lambda i.(r_i) \text{ is computable.}^6 \tag{8}$$

(i) *We say* $\mathbf{r}$ *is* primitive recursive $\stackrel{\text{def}}{\Leftrightarrow}$ *the function*

$$\lambda i.(r_i) \text{ is primitive recursive.} \tag{9}$$

Definition *We associate with each real* $\mathbf{r}$ *between* $\mathbf{0}$ *and* $\mathbf{1}$ *inclusive a corresponding function* $g_{\mathbf{r}}$ *thus. For each* $x \in N$, *let*

$$g_{\mathbf{r}}(x) \stackrel{\text{def}}{=} \begin{cases} 1, & \textit{if, in the binary expansion of } \mathbf{r}\textit{, there is a} \\ & \text{consecutive } \textit{run of 1's of length} \\ & \text{exactly } x \text{ and } \textit{with a 0 on } \text{each} \\ & \textit{side of the run;} \\ 0, & \textit{otherwise.} \end{cases} \tag{10}$$

For example, if $\mathbf{r} = .010110111011110\cdots$, then $g_{\mathbf{r}}$ is 1 at all arguments > 0 and is 0 at argument 0. If *instead* $\mathbf{r} = .10111000000000\cdots$, then $g_{\mathbf{r}}$ is 1 at arguments 0 and 3 and it's 0 elsewhere. For this *latter* example, the initial 1 in the binary expansion does *not* make $g_{\mathbf{r}}(1) = 1$ since this initial 1 does *not* have a 0 on *each* side. $g_{\mathbf{r}}(0) = 1$ since, for example, the 6th and 7th digits are 0's and have a block of 0 1's between them.

a. (20 points)

Prove there is a *computable* real $\mathbf{r}$ such that $g_{\mathbf{r}}$ is *not* computable.

Hint for C3a: You may establish computability by exhibiting an informal algorithm (and showing totality), and you may use without proof any standard text book results about r.e. sets *provided you say which result(s) you are using where.*

b. (5 points)

Prove there is a *primitive recursive* real $\mathbf{r}$ such that $g_{\mathbf{r}}$ is *not* computable.

Hint for C3b: You may use without proof any of the standard text book closure properties for the class of primitive recursive functions *provided you say which closure properties you are using where.* You may also use without proof any standard text book examples of primitive recursive functions, *but you must say what they are.* Again you may cite without proof suitable results about r.e. sets.

[5] For example, the real number $\frac{1}{2}$ (in binary) is $.1000000000\cdots$, the real number $\frac{2}{3}$ (in binary) is $.1010101010\cdots$, and the real number $\mathbf{1}$ (in binary) is $.1111111111\cdots$. To see the latter two, just sum corresponding infinite geometric series.

[6] Here we use Church's lambda-notation to name a function defined by a formula (as is standard in Lisp). For example, $\lambda x, y.(x+y)$ denotes the function that maps each pair of non-negative integer x, y to their sum, and $\lambda i.(r_i)$ denotes the function of one non-negative integer i argument mapping it to the bit r_i.

C4 Theory (25 points)
Fix a standard programming formalism φ for computing all the *one-argument* partial computable functions which map the non-negative integers into themselves. Code (Gödel) number the φ-programs *onto* the entire set of non-negative integers. Let φ_p denote the partial function computed by program (number) p in the φ-system. You may assume without proof that, in the φ-system, Universality, S-m-n, and the Kleene Recursion Theorem (KRT) hold. *You may not need all these assumptions — just say which ones you are using where.* Let $W_p \stackrel{\text{def}}{=}$ the domain of φ_p.[7]

a. (10 points)
Prove there *is* a computable function f such that

$$\{W_{f(x)} \mid x \in N\} = \text{the collection of all finite subsets of } N. \tag{11}$$

Hint for C4a: You may establish computability by exhibiting an informal algorithm (and showing totality).

b. (15 points)
Prove there is *no* computable function g such that

$$\{W_{g(x)} \mid x \in N\} = \text{the collection of all infinite r.e. subsets of } N. \tag{12}$$

Hint for C4b: Suppose for contradiction otherwise. Create an *infinite r.e.* set A differing from each of $W_{g(0)}$, $W_{g(1)}$, $W_{g(2)}$, $\ldots$. To do this think *in*formally about how to enumerate algorithmically such an A from dovetailed algorithmic enumerations of $W_{g(0)}$, $W_{g(1)}$, $W_{g(2)}$, $\ldots$. Make sure you list into A an element from each $W_{g(i)}$ *and* that you do *not* list into A at least one element from each $W_{g(i)}$. Be sure to *show why this is possible.*

[7] Then $W_0, W_1, W_2, \ldots$ provides a standard listing of *all* the r.e. sets (of non-negative integers).