C1 **Theory** (25 points)

a. (6.25 points)
Show that

$$L_1 = \{w \in \{a,b\}^* \mid w \text{ contains subword } aab \textit{ or } \text{20th from last symbol of } w \text{ exists } \& = b\} \quad (1)$$

is regular. You may use with*out* proof any standard text book results about regular sets *provided you clearly say which results you are using when.*[1]

b. (6.25 points)
Find a *deterministic* finite automaton $\mathcal{M}'$ which accepts the same language (over $\{a,b\}$) as the non-deterministic finite automaton $\mathcal{M}$ depicted in table form just below.

δ	a	b
`start` 1	$\{1,2\}$	$\{1\}$
2	$\{3\}$	$\{3\}$
3	$\{4\}$	$\{4\}$
4	$\{5\}$	$\{5\}$
`final` 5	$\emptyset$	$\emptyset$.

c. (6.25 points)
Employ an appropriate pumping lemma to show that

$$L_2 = \{a^m b^n \mid m \text{ is a perfect square } \vee \ n \text{ is odd}\} \quad (2)$$

is *not* regular.

d. (6.25 points)
Employ an appropriate pumping lemma to show that

$$L_3 = \{ab^p \mid p \text{ is prime}\} \quad (3)$$

is *not* context free, i.e., is *not* accepted by any push down automaton.

[1] L_1 is *not* a standard text book regular language. (⌣̈)

C2 **Theory** (25 points)

a. (12.5 points)
Consider the following finite automaton $\mathcal{M}$ expressed in tabular form.

δ	a	b
start 1	2	4
2	3	4
final 3	3	3
4	2	5
5	2	6
6	6	6.

This $\mathcal{M}$ *is* minimal state (for the accepting task it performs).
Explicitly employ Myhill-Nerode to *prove* this $\mathcal{M}$ is minimal state.

Hint: You may find it useful to draw the state diagram of $\mathcal{M}$.
Find a relevant spanning S by considering how to reach each state of $\mathcal{M}$ from its start state. Show this S can't be reduced in size and still be relevantly spanning. The number of combinations of six things taken two at a time is 15.

b. (12.5 points)
Explicitly program a *deterministic* push-down automaton which accepts all and only the strings in

$$L = \{a^n b^{2n} \mid n > 0\}. \tag{4}$$

Up to half credit if your pda is not deterministic.

C3 **Theory** (25 points)

Let $N =$ the set of non-negative integers.

We write $(f_i \mid i \in N)$ for the infinite sequence of functions $(f_0, f_1, f_2, \ldots)$.

Definition *A sequence of functions* $(f_i \mid i \in N)$ *is said to be* uniformly computable $\stackrel{\text{def}}{\Leftrightarrow}$ *the function* $\lambda i, x \,\text{.}\, f_i(x)$ *is computable.*

Example 1 *For each* i, x*, let*

$$f_i(x) = i^2 x^3 + 4i. \tag{5}$$

Then this $(f_i \mid i \in N)$ is *clearly uniformly computable.*

Definition *A sequence of functions* $(f_i \mid i \in N)$ *is* uniformly primitive recursive $\stackrel{\text{def}}{\Leftrightarrow}$ *the function* $\lambda i, x \,\text{.}\, f_i(x)$ *is primitive recursive.*

Example 2 *Define* $\lambda i, x \,\text{.}\, f_i(x)$ *as in (5) of Example 1 above. Then* $(f_i \mid i \in N)$ is*, in fact, uniformly* primitive recursive*.*

a. (12.5 points)

Prove, employing the Hint just below that there is a sequence of functions $(F_i \mid i \in N)$ such that

1. $(\forall i)[F_i$ is computable$]$ *and*
2. $(F_i \mid i \in N)$ is *not* uniformly computable.

Hint for C3a: Let A be an r.e. *not* computable set. Write A as $\{a_0 < a_1 < a_2 < \ldots\}$. For each i, x, let $F_i(x) \stackrel{\text{def}}{=} a_i$.
Show that, for *each, fixed* $i \in N$, $\lambda x \,\text{.}\, F_i(x)$ is a primitive recursive (hence, computable) function.
Suppose for contradiction $(F_i \mid i \in N)$ *is* uniformly computable. Then $\lambda i, x \,\text{.}\, F_i(x)$ is computable.
Show, then, that $\lambda i \,\text{.}\, F_i(0)$ is computable, monotone increasing, and has range A.
Show how to obtain a contradiction from this.

b. (12.5 points)

Prove, employing the Hint just below that there is a sequence of functions $(G_i \mid i \in N)$ such that

1. $(\forall i)[G_i$ is primitive recursive$]$,
2. $(G_i \mid i \in N)$ is *not* uniformly primitive recursive, and
3. $(G_i \mid i \in N)$ *is* uniformly computable.

Hint for C3b: Fix a standard algorithmic coding of the finite sets of equations each defining a one argument primitive recursive function 1-1 onto N. Let G_i be the one argument primitive recursive function defined by the finite set of such equations with code number i. Do *not* waste time providing details about such a coding.
Trivially, $(\forall i)[G_i$ is primitive recursive$]$.
Suppose for contradiction $(G_i \mid i \in N)$ is uniformly primitive recursive. Hence, $\lambda i, x \,\text{.}\, G_i(x)$ is primitive recursive. Define $g(x) = 1 + G_x(x)$. To get a contradiction, *show that* g *is both primitive recursive and not primitive recursive.*
Argue **very informally and briefly** that $\lambda i, x \,\text{.}\, G_i(x)$ is computable.

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. Let $W_p \stackrel{\text{def}}{=}$ the domain of φ_p.[2] You may assume with*out* proof that, in the φ-system, Universality, S-m-n, *and the Kleene Recursion Theorem (KRT)* hold.

As usual: $\downarrow$ means 'is defined'; and $\uparrow$ means 'undefined'.

Explicitly employ the hint further below to prove the following theorem.

Theorem For each non-negative integer x, let

$$\psi(x) = \begin{cases} \text{the } \textit{least } y \in W_x \ , & \text{if } W_x \neq \emptyset; \\ \uparrow, & \text{otherwise.} \end{cases} \tag{6}$$

Then ψ is *not* partial computable.

Hint: Suppose for contradiction otherwise.

Employ KRT to obtain a φ-program e such that (7), (8), and (9) below *each* hold.

$$1 \in W_e \subseteq \{0, 1\}. \tag{7}$$

Note that (7) will force $\psi(e)\downarrow \in \{0, 1\}$.

$$\psi(e) = 1 \Rightarrow 0 \in W_e. \tag{8}$$

$$\psi(e) = 0 \Rightarrow 0 \notin W_e. \tag{9}$$

Finally show that the behavior of your e *is* contradictory.

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