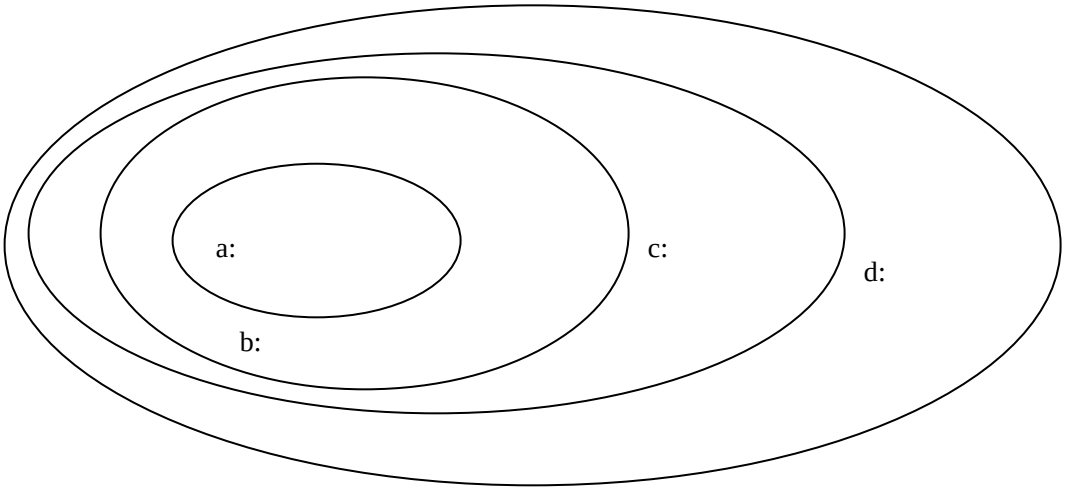
Theory of Computation
Practice Final

1. **True/False questions**: For each part, circle either True or False. (23 points: 1 points each)

a.	A TM can compute anything a desktop PC can, although it might take more time.	True	False
b.	A Push-Down Automata can compute things that a TM cannot compute.	True	False
c.	Every Turing-decidable language is also Turing-recognizable.	True	False
d.	The Halting problem is decidable.	True	False
e.	All problems have an algorithm that will solve/decide them.	True	False
f.	$4n^2 = O(n)$	True	False
g.	$3n^3 = O(n^3)$	True	False
h.	$n\log n = o(n^2)$	True	False
i.	$2^n = O(n^{20})$	True	False
j.	You <u>cannot</u> build a DFA to recognize $\{0^{500}1^{10000} \cup 1^{1000}0^{200}\}$	True	False
k.	NP is the class of languages with polynomial time verifiers.	True	False
l.	The various sorting algorithms (e.g., bubblesort, heapsort) are in NP	True	False
m.	Most theoretical computer scientists believe that P = NP.	True	False
n.	All languages are Turing-recognizable	True	False
o.	The class of regular languages is closed under union	True	False
p.	All languages are decidable	True	False
q.	A regular language L may not be context-free.	True	False
r.	A_{DFA} is decidable. A_{DFA} = {<B,w>| B is a DFA that accepts input string w}.	True	False
s.	Deterministic and non-deterministic PDA's have equivalent expressive power.	True	False
t.	If a problem A is reducible to problem B, then problem A must be no harder than B.	True	False
u.	An algorithm implemented on a single tape Turing machine will always have the same running time (e.g., big-O value) when run on a 2-tape Turing machine.	True	False
v.	NP is the class of languages decided by some nondeterministic polynomial time Turing Machine.	True	False
w.	From a computability perspective, every multi-tape Turing machine has an equivalent single-tape TM.	True	False

2. **Short answer questions.** Answer each question in a few sentences. (14 points: 2 each)

   a. The diagram below show a hierarchy of the languages we learned, with respect to computability. Write the proper language next to the labels a-d in the diagram below such that the hierarchy is correct. The languages are: Turing-recognizable, regular, decidable, context-free.



   b. A finite automata will run until its input is completely processed and then it will stop. This is not true for a Turing machine. Explain why.

   c. A language is Turing-recognizable if some Turing machine recognizes it (this is a definition). But what does it mean when we say that a TM recognizes a language? The answer can be quite simple (one sentence) but please be precise.

d. A language is Turing-decidable if some Turing machine decides it. What does it mean for a Turing machine to decide a language? Again, please be precise, but you can be relatively informal.

e. We are given a problem and find out that it is undecidable. Could there be an algorithm to solve it in polynomial time? Answer “yes” or “no” and then explain/justify your answer.

f. I tell you that an algorithm runs in $O(2^n)$ but yet is in P. How can this be?

g. Assume that someone finds a polynomial time solution to an NP-complete problem. 1) What does that say about all NP-complete problems? 2) What does that say about the question of whether P = NP?

3. (7 points) Draw the DFA that, for the alphabet $\Sigma = \{0,1\}$ accepts all strings that do not have any consecutive 0's or 1's (e.g., 0, 0101, 101, and 10 are accepted but 11, 001, and 01001 are not)

4. (7 points) Provide a Context Free grammar that generates the language 00*1*.

5. (7 points) Let $E_{DFA} = \{<A> | A \text{ is a DFA and } L(A) = \varnothing\}$. Prove that E_{DFA} is decidable by providing an algorithm that decides it. You should also comment on why the provided algorithm is decidable.

6. (10 points total) Give implementation level descriptions of Turing machines that decide the following languages over the alphabet {a,b,c}. Recall that implementation level is lower level than the pseudo code that we use to describe algorithms (at the implementation level you talk about scanning the tape and tape movements).

   {w| w contains more than 5 times as many b's as a's}

7. (5 points) Let A_{DFA} = {<B,w>| B is a DFA that accepts input string w}. Provide a simple explanation (one could call it a very informal proof) for why this language is decidable.

8. (9 points) Describe a PDA that accepts the following languages. For part b, if you want, rather than describing it from scratch, you can just say how you would modify your answer from part a. Note: part b is a bit tricky. Hint: for part b, you will need to utilize a capability of the PDA not used in part a.

    a. $L = \{0^m1^n: n \leq m\}$