Midterm

Please submit by 3:30PM. Slide the paper copy under my door if I’m not around, or email me the solution.

Each question carries 4 points. For each question, you must provide an option and a two-sentence explanation of why you chose this. No points without an explanation. Nothing more than two sentences please, just give me the core of your argument. If you end up at the wrong answer but have some explanation that is meaningful, I might give some (at most 2) points.

No searching the Internet or talking to others. Refer to your notes and books freely. Answers in blue.

1. Let $S$ contain all finite sets of $\mathbb{N}$ (natural numbers). Which is true?
   
   (a) There is a bijection between $\mathbb{N}$ and $S$.
   (b) There is a bijection between $\mathbb{R}$ and $S$.
   (c) Neither of the above is true.

   Any finite set $S$ can be represented as a finite string $s_1, s_2, \ldots, s_k$. Representing this as binary, we uniquely map $S$ to a natural number.

2. My company makes Turing Machines, and I need to fund the right research to make things faster. To convince the investors, I need to get the largest asymptotically faster speedup for problems (a constant factor speedup isn’t worth the money). What should I fund?

   (a) Hardware that works on a $k$-symbol alphabet, instead of just binary.
   (b) Increasing the number of tapes as much as possible.
   (c) Laying out the memory/tapes in a two-dimensional format.

   Changing the alphabet at best gives a constant factor speedup. With two tapes, we can simulate any number of tapes with at most logarithmic overhead (Hennie-Stearns theorem). The best simulation of a two-dimensional tape TM by standard TMs has quadratic blowup.

3. Remember that FACTORING = $\{\langle N, k \rangle \mid N$ has non-trivial factor less than $k \}$. Suppose I proved that FACTORING is $\mathbb{NP}$-complete. What could this imply?

   (a) $\mathbb{P} = \mathbb{NP}$.
   (b) $\mathbb{NP} = \text{co-}\mathbb{NP}$.
   (c) This is more evidence of the hardness of FACTORING.
   (d) This is not possible, since it contradicts our current knowledge.

   FACTORING $\in \text{co-}\mathbb{NP}$, and if any co-$\mathbb{NP}$ problem is $\mathbb{NP}$-complete, $\mathbb{NP} = \text{co-}\mathbb{NP}$. (First problem of Assn 2.)
4. Suppose I proved FACTORING $\not\in \mathbb{P}$. It turns out that the NSA just built a new quantum computer using gravitational waves that solves FACTORING. It runs in time polynomial in input size!

(a) This proves that $\mathbb{P} = \mathbb{NP}$.
(b) This shows that the Church-Turing thesis is wrong.
(c) This shows that the extended Church-Turing thesis is wrong.
(d) This is bogus. There is a contradiction in the question.

This new machine solves FACTORING in polynomial time, though no Turing machine can do the same. Thus, this new machine cannot be simulated on a Turing Machine with polynomial overhead.

5. Define the NO-ENEMY problem: I want to invite people to a party. But some pairs are (mutual) enemies, and I don’t want to have my party to have such pairs. Can I invite at least $k$ people?

Define the ONE-ENEMY problem: Same party setting but I want to call people according to a different rule. Every invited person has exactly one enemy who is also invited. Can I invite at least $k$ people?

I have an algorithm that converts, in polynomial time, an instance of the ONE-ENEMY problem to an instance of the NO-ENEMY problem. What does this imply?

(a) $\mathbb{P} = \mathbb{NP}$.
(b) $\mathbb{NP} = \text{co-\mathbb{NP}}$.
(c) Nothing, this is already implied by current knowledge.

NO-ENEMY is Independent-Set, ONE-ENEMY is maximum matching, and both are in $\mathbb{NP}$. NO-ENEMY is $\mathbb{NP}$-complete, and we already know that all $\mathbb{NP}$ problems can be reduced to it.

6. Define the POL-ONE-ENEMY variant. Similar to ONE-ENEMY, but people are tagged Republicans and Democrats. No two people in a party are enemies. Can I invite at least $k$ people, such that each invited person has exactly one enemy who is also invited?

I have an algorithm that converts, in polynomial time, an instance of the NO-ENEMY problem to an instance of the POL-ONE-ENEMY problem. Implications?

(a) $\mathbb{P} = \mathbb{NP}$.
(b) $\mathbb{NP} = \text{co-\mathbb{NP}}$.
(c) Nothing, already implied by current knowledge.

POL-ONE-ENEMY is bipartite matching which is in $\mathbb{P}$. By reducing an $\mathbb{NP}$-complete problem to $\mathbb{P}$, we prove $\mathbb{P} = \mathbb{NP}$. (As an aside, ONE-ENEMY is also in $\mathbb{P}$, though we didn’t discuss that in class.)

7. Define $\mathbb{QP}_1 = \bigcup_{c \geq 1} DTIME(n^{c\log n})$. Define $\mathcal{L} \leq_q \mathcal{L}'$ if one can reduce $\mathcal{L}$ to $\mathcal{L}'$ using an algorithm that runs in time $n^{c\log n}$ (for some constant $c$). Which is false?

(a) If $\mathcal{L} \leq_q \mathcal{L}'$ and $\mathcal{L}' \leq_q \mathcal{L}''$, then $\mathcal{L} \leq_q \mathcal{L}''$. 

(a) If $\mathcal{L} \leq_q \mathcal{L}'$ and $\mathcal{L}' \leq_q \mathcal{L}''$, then $\mathcal{L} \leq_q \mathcal{L}''$. 

2
(b) For all $L \in \text{NP}$, $L \leq_q \text{SAT}$.

(c) $\text{NP} \subseteq \text{QP}_1$.

Let $f$ be reduction from $L$ to $L'$ and $g$ be reduction from $L'$ to $L''$. Note that $|g(f(x))|$ could be as large as $|f(x)|^{c \log |f(x)|} = (n^{c' \log n})^{c \log(n^{c' \log n})} = n^{\Omega(\log^3 n)}$. This does not give a $\text{QP}_1$ reduction.

8. Define $L^2 = \text{SPACE}(\log^2 n)$. I just proved an awesome theorem! A nice hierarchy: $\text{NL} \subseteq L^2 \subseteq \text{QP}_1$. Advice?

(a) Go and publish this! This is a new result.

(b) Go and check this! There must be a bug, it contradicts existing work.

(c) Go and read! It follows from existing work.

$\text{NL} \in L^2$ by Savitch’s theorem. $L^2 \subseteq \text{QP}_1$ because $\text{SPACE}(s(n)) \subseteq \text{DTIME}(s(n)^2)$. 