CSEP531 Homework 6 Solution

1. (a) UNKNOWN. If $P = NP$ then it is true; otherwise it is false.

   (b) TRUE. Any problem in P is in NP, thereby being reducible to the NP-complete problem HAMPATH. (Directly, here is a legitimate "reduction": run the polynomial time algorithm for PATH, if the answer is yes, produce a cycle as an instance of HAMPATH; otherwise produce a graph with no edges as an instance of HAMPATH).

   (c) FALSE. The language Σ^* is decidable. Its intersection with any language L is L itself. So if we choose L to be a language that is recognizable but not decidable - such as the language HALT which contains (the description of) all machines that halt - then its intersection with Σ^* is not decidable.

   (d) UNKNOWN. If $P = PSPACE$ then it is true; otherwise it is false. Because this one is trickier than we had intended, we also gave full credit for the answer false (since, this language is in LOGSPACE, and we know by the space hierarchy theorem that LOGSPACE is strictly contained in PSPACE).

   (e) UNKNOWN. Well, you know you don't know it, don't you?

   (f) FALSE. Assume that such L exists, then we have a map f such that $x \in L$ iff $f(x) \notin L$. Furthermore, L is recognized by some machine M. We can build a decider for L by running M on x and $f(x)$ simultaneously. One of them has to halt. From the answer of the run that halts first, we can deduce if x belongs to L or not.

   (g) TRUE. All problems in P, e.g. shortest path and minimum spanning tree, are also in NP.

   (h) TRUE. The reduction is the same as in (b).

   (i) UNKNOWN. The reason is the same as in (a).

   (j) TRUE. This is a part of the definition of NP-completeness.

   (k) UNKNOWN. We still don't know if factoring, which is both in P and $coNP$, is in P – it being in P would be a very bad news for our already not-too-good economy.

   (l) TRUE. A polynomial time reduction uses polynomial space. Therefore, we can solve any instance of X by first reducing it to an instance of Y and then solving that instance.

2. First, note the following inclusions: $P \subseteq NP \cap coNP \subseteq NP \subseteq PSPACE \subseteq DECIDABLE \subseteq RECOGNIZABLE$ where $DECIDABLE$ and $RECOGNIZABLE$ are class of languages/problems that are decidable and recgonizable respectively. So, if a problem is known to be in a class, it is in all of its supperclasses and if it is known not to be in a class, it is not in any of its subclasses. Also, if a problem is NP-complete, then it is in NP.

   (a) TRUE: recognizable; FALSE: decidable (thus, everything else). We covered both of these in class.

   (b) TRUE: in P (thus, everything else except NP-complete), since simulating the machine for $|w|$ steps clearly takes polynomial time. UNKNOWN: NP-complete. If $P = NP$, then it is also NP-complete; otherwise, it is not.

   (c) TRUE: decidable (thus recognizable too). FALSE: in P. UNKNOWN: everything else. Clearly, we can simulate the machine for $2^{|w|}$ steps. On the other hand, we know this is one of the problems that are not in P due to the time hierarchy theorem.

(d) FALSE: recognizable (thus, everything else). Note that its complement is recognizable. Therefore, if it is also recognizable, we can build a decider by running the two machines on x simultaneoulsy. One of the them has to halt. We can deduce if x is in the language or not from the answer of the machine that halts first.

(e) TRUE: NP-complete (thus, is in NP and all of its superclasses). UNKNOWN: in P and $NP \cap coNP$. This is 3-SAT, so the answer should be clear.

(f) TRUE: in P (thus is in all of its superclasses). UNKNOWN: NP-complete. If $P = NP$, then it is; otherwise, it is not.

(g) TRUE: in $NP \cap coNP$ (thus is in all of its superclasses). UNKNOWN: in P and NP-complete. See question 1(k).