CSL 705 Theory of Computation
Minor 2, Sem II 2014-15, Max 40, Time 1 hr

Name ___________________________ Entry No. ___________________________ Group

Note (i) Write your answers neatly and precisely in the space provided with each question including back of the sheet. You won't get a second chance to explain what you have written.
(ii) You can quote any result covered in the lectures without proof but any other claim should be formally justified.

1. Which of the following statements are true. Justify briefly. (5 × 4)

   (a) If \( L \in PSAPCE \) then \( L_{QBF} \leq poly-space \) \( L \). \( L_{QBF} \) corresponds to Quantified Boolean Formula. True.

   For any non-trivial language \( L \), we have \( x_1 \in L \) and \( x_2 \notin L \). Let us define a Turing computable function \( f : \Sigma^* \Rightarrow \Sigma^* \) such that if \( x \in L_{QBF} \) then \( f(x) = x_1 \) else \( f(x) = x_2 \). This function is computable in polyspace since \( L_{QBF} \in PSPACE \).

   (b) If the unsatisfiability problem \( L_{USAT} \in NP \) then \( NP = co-NP \). True.

   We know that \( L_{USAT} \) is co-NP complete under polytime reduction. So if \( L_{USAT} \in NP \) then \( \forall L \in co-NP \) there exists a NP algorithm for \( L \). First reduce \( L \) to \( L_{USAT} \) and subsequently run the NP algorithm for \( L_{USAT} \). So \( L \in NP \).

   (c) If \( L_{SAT} \in NL \) then \( NP = P. \) NL is non-deterministic logspace. True.

   If \( L \in NL \) then \( L \in P \). This is using the same argument that if \( L \in DSPACE(f(n)) \) then \( L \in DTIME(2^{O(f(n))}) \). Hence \( L_{SAT} \in P \) and so \( P = NP \).

   (d) NPSPACE is closed under complementation, i.e. if \( L \in NPSPACE \) then \( \bar{L} \in NPSPACE \). True.

   We know that if \( L \in NPSPACE(f(n)) \) for \( f(n) \geq \log n \), then \( L \in PSPACE((f(n))^2) \). If \( f(n) \) is polynomial so is \( (f(n))^2 \). If \( f(n) \) is \( o(\log n) \), then \( L \in \log n \), so it still holds. Since \( PSPACE \) is closed under complementation \( \bar{L} \in PSPACE \) and consequently in \( NPSPACE \).
2. The language $L_{5SAT}$ is the set of satisfiable boolean formula in CNF with exactly 5 literals per clause. Prove that $L_{5SAT}$ is NP complete. (8) Since $L_{SAT} \in NP$, $L_{5SAT} \in NP$.

We will show that $L_{3SAT} \leq polytime L_{5SAT}$. Consider a clause $C_i = (y_1 \lor y_2 \land y_3)$ where $y_i \in \{x_1, \bar{x}_1, x_2, \bar{x}_2, \ldots, \bar{x}_n\}$. We construct the following 5CNF for this $(y_1 \lor y_2 \lor y_3 \lor a \lor b)$ where $a, b \in z_1, \bar{z}_1, z_2, \bar{z}_2$ for new boolean variables $z_1, z_2$ - denote these 4 clauses by $C_{i,1}, C_{i,2}, C_{i,3}, C_{i,4}$. It can be shown that

$$C_i \equiv C_{i,1} \land C_{i,2} \land C_{i,3} \land C_{i,4}$$

Moreover if there are $m$ clauses and $n$ variables in the 3CNF formula, then there are at most $4m$ clauses in the new formula and $n + 2m$ variables in the new 5 CNF formula, i.e., it is polynomial in size of the original formula.

3. The language $BH$ (Bounded Halting problem) is defined as follows.

$$L_{BH} = \{ \langle x, \alpha, 1^n, 1^t \rangle | \exists u, |u| = n \text{ } M_\alpha \text{ accepts } \langle x, u \rangle \text{ in } t \text{ steps } \}$$

Here $M_\alpha$ is a non-deterministic TM with code $\alpha$ and $u$ is a set of non-deterministic choices that $M_\alpha$ makes on input $x$. Once $u$ is fixed then the machine behaves like a deterministic machine.

(i) Show that $L_{BH}$ is an NP complete language. (8)

$L_{BH} \in NP$ since there exists a multitape TM $M$ that can simulate $M_\alpha$ in time polynomial in $n + t + |x| + |\alpha|$. It copies $x$ into one of its tapes and simulates the moves of $M_\alpha$. It guesses the non-deterministic moves of $M_\alpha$ in $O(n)$ steps and then simulates each of the $t$ moves of $M_\alpha$ in $O(\alpha)$ steps. So the total time is $|x| + O(n) + |\alpha| \cdot t$ which is polynomial in input size.

(If $t$ and $n$ were not given as unary then $t$ is not polynomial in $\log t$ that answers the second part).

For any $L \in NP$, there exists an $M_L$ that accepts an $x \in L$ in non-deterministic polynomial time say $p(|x|)$ of which some or all the moves use non-deterministic choices. The choices can be represented as a string $y = a_1a_2\ldots a_n$ where $a_i$ is at most $\log |\alpha|$ and $\alpha$ is the code of $M_L$, $n \leq p(|x|)$. So we can form the tuple $\langle x, \alpha, 1^n, 1^{p(|x|)} \rangle$. The length of tuple is polynomial in $|x|$ - the code $\alpha$ is a fixed constant independent of $|x|$. For completeness, we also note that $p(|x|)$ 1s can be written out in polynomial time, say by Horner’s rule.

(ii) What is the significance of writing $1^n$ and $1^t$? (4)