Homework 2, (carries no grade - solutions will be posted by 25th)

1. (20 points, compulsory)

   (i) Classify each of these languages as regular (in $\mathcal{R}$), context free but not regular, (in $\mathcal{C} \setminus \mathcal{R}$) or not context free (not in $\mathcal{C}$). In all cases, justify your answer.
   (a) $\{w : w \neq xx^{reverse}$ for any $x \in \Sigma^*\}$.
   (b) $\{w : w = xx$ for some $x \in \Sigma^*\}$.
   (c) $\{w : w \neq xx$ for any $x \in \Sigma^*\}$.

   (ii) Consider the grammar:

$$S \to aSd \mid S_1$$

$$S_1 \to S \mid S_2$$

$$S_2 \to bS_2c \mid \epsilon$$

   (a) What is the language derived by the above grammar (describe as succinctly as possible)?
   (b) Give a Chomsky normal form for the language - derived from the above grammar

2. (20 points, compulsory) Are the following languages context free? In all cases, justify your answer.

   (i) $\{a^n b a^m : m \neq n; n, m \in N\}$
   (ii) $\{a^n b^n c^m d^m : n, m \in N\}$.
   (iii) $\{a^n b^m c^n d^m : n, m \in N\}$.
   (iv) $\{a^n b^{2n} a^n : n \in N\}$.
   (v) $\{a^{n^2+3} : n \in N\}$

3. (20 points, compulsory) In all cases, justify your answer.

(i) Let C be the class of context-free languages, and R the class of regular languages. Let

$$RC =_{def} \{L_1 \setminus L_2 : L_1 \in R, L_2 \in C\},$$

and

$$CR =_{def} \{L_1 \setminus L_2 : L_2 \in R, L_1 \in C\}.$$

Prove or disprove (to disprove, give counterexample languages):
(a) $RC \subseteq C$.
(b) $CR \subseteq C$.

(ii) If L is context free, is $1/2L$ always context free? (see definition in bonus question of HW1–you do not need to have done the bonus question of HW1 to answer this one)

(iii Is the class of context free languages closed under the REFLECT operation of HW1?

(iv) Is the class of context free languages closed under the INV-REFLECT operation of HW1?

(v) If L is regular, then is $REFLECT(L)$ always context free? Hint: one nice way is to use the regular grammar for L defined in class to get a CFG for $REFLECT(L)$

4. (20 points compulsory) This is really two questions. Which of $\mathcal{C}$ and $\mathcal{R}$ are closed under the *Shuffle* operation (either one or both)? Informally, the *Shuffle* operation takes every string from L_1 and every string from L_2, interleaves them in every possible way, and throws the result into *Shuffle*(L_1, L_2). Formally,

$$\textit{Shuffle}(L_1, L_2) =_{def} \{a_1 b_1 \ldots a_k b_k : a_1 a_2 \ldots a_k \text{ is a string in } L_1 \text{ and}$$

$$b_1 b_2 \ldots b_k \text{ is a string in } L_2\}.$$

Here a_i and b_i are arbitrary strings, and could be empty.

5. (20 points, compulsory)

(i) Is there an enumeration of algorithms?, i.e, is there a computable (an algorithm that computes a) bijection from the natural numbers to the set of all algorithms (always halting pseudocodes)? Why?

(ii) Discuss the merit of the argument: since all realistic computers today have some absolute bound on their memory, the halting problem for pseudocodes (running on these computers) could be solved. Is it true that the halting problem for machines that have a finite memory is decidable? Why (not)?

6. (BONUS 20 pts, not compulsory) Show one of the following.

   (i) Show that the language $= \{0^n 1^n : n \geq 1\} \cup \{0^n 1^{2n} : n \geq 1\}$ does not have a deterministic pushdown automaton. Hint: Intuitively, a DPDA should have to erase almost everything from the stack to recognize the first part, so it won't be able to recognize the second part. Another way to think about this is to try to prove a pumping lemma for DPDA's and apply it.

   (ii) $= \{0^j 1^k 2^l : j = l\} \cup \{0^j 1^k 2^l : j = k\}$ is inherently ambiguous.

   Note: as mentioned in class, by showing a language to be inherently ambiguous you are also showing that it has no DPDA – but *not* viceversa. There are some languages that have no DPDA's but have unambiguous grammars.