Problem set 12 - non-uniform computation

This exercise contains a few questions on uniform vs. non-uniform computation for those of you who haven’t seen it before.

Notation for this exercise: We say a uniform TM machine $M(x; \cdot)$ accepts a language $L$ with non-uniform advice $\{a_n\}$, if for every $x \in \{0,1\}^*$, $x \in L$ iff $M(x, a_{|x|}) = 1$. Notice that the advice depends only on the input length and not on the specific input itself.

We say $L \in DTime(t(n))|f(n)$, if there exists a uniform machine $M(x; \cdot)$ in $DTime(t(n))$ that accepts $L$ with non-uniform advice $\{a_n\}$ and $|a_n| \leq f(n)$. I.e., on inputs of length $n$, $M$ gets $f(n)$ bits of advice (that depend on the input length only) and solves $L$ in deterministic time $t(n)$. Define

$$ P|Poly = \bigcup_{k_1, k_2} DTIME(n_1^{k_1})|n_2^{k_2} $$

Let $Size(s(n))$ denote the set of languages $L$ that can be solved by a non-uniform family of circuits $\{C_n\}$ of size $s(n)$.

1. Prove that $P|Poly = Size(poly)$.

2. Find a language $L$ that cannot be solved by any uniform TM, but can be solved by circuits of size 1 (i.e., in $DTIME(1)|1$).

3. Prove that $BPP \subseteq P|Poly$.

4. Prove that if $NP \subseteq P|\text{log}$ then $NP = P$