Name ______________________ Student No. ______________________

No aids allowed. Answer all questions on test paper. Use backs of sheets for scratch work.

Total Marks: 100

[25] 1. Show that a 2-PDA (a PDA with two stacks) can simulate any Turing machine. Give a high-level argument, outlining the key ideas. Make sure your answer mentions the notion of a *configuration*.

Solution: The idea is that a 2-PDA can keep track of the configuration vq_iw of the TM by keeping v, in reverse, in the first stack, and w in the second stack.

[25] 2. Explain the notion of *robustness* of the Turing machine model. Illustrate it with the following variant of a Turing machine: it is allowed to have k tapes, and all k tapes are infinite in both directions. Show that a Turing machine with a single tape, infinite in one direction only, can simulate such an extended machine.

Solution: Reasonable variants of a TM still decide the same languages. In the case of k tapes, simulate the tapes on a single tape by concatenating their contents, and doing a pass to first establish where the original heads were (mark symbols that are being scanned in a special way, say, $\dot{a}$), and then a second pass to make the appropriate changes. When a head wants to move left of the beginning of its portion of the tape, move all squares one position to the right.

[25] 3. Recall that $L^* = \{w | w = w_1 w_2 \dots w_k, k \geq 0, w_i \in L\}$. Show that the class of Turing-recognizable languages is "closed under $*$" (i.e., show that if L is Turing-recognizable, so is L^*). Show, however, that they are not closed under complementation (you may assume that $A_{\text{TM}} = \{\langle M, w\rangle |\ M \text{ accepts } w\ \}$ is not decidable).

Solution: Nondeterministically guess the division of w into substrings $w_1 w_2 \dots w_k$. Simulate the machine M for L on w_i, and move over to w_{i+1} if it accepts. Accept if all substrings are accepted.

[25] 4. Consider the language $EQ_{\mathrm{DFA}} = \{\langle D_1, D_2\rangle | L(D_1) = L(D_2)\}$. Show that it is possible to decide EQ_{DFA} by examining a finite set of strings. (**Hint.** If $L(D_1) \neq L(D_2)$, there is a string s on which they differ; consider the shortest such string; how short is it?)

Solution: For DFAs D_1, D_2, $L(D_1) = L(D_2)$ iff D_1, D_2 accept the same strings up to length $m \cdot n$, where $m = |Q_{D_1}|$ and $n = |Q_{D_2}|$. One direction of this claim is trivial. Suppose that $L(D_1) \neq L(D_2)$. We show then that they differ on some string s of length at most $n \cdot m$. For any given s, let $\{q_1, \ldots, q_{|s|}\}$ and $\{r_1, \ldots, r_{|s|}\}$ be the sequences of states that D_1, D_2 visit on s, respectively. If $|s| > m{\cdot}n$, then by the pigeon hole principle there is a repeated pair if states, i.e., $(q_i, r_i) = (q_j, r_j)$, and so s can be shortened, and still "separate" D_1, D_2.