there is no DPDA M such that $L(M) = B$, is beyond the scope of this course.) (21 pts. total, for 84 total on the problem set)

Answer: L_1 equals the set of strings that have an odd number of a's, since the "a" in "xay" can be taken as the middle a in such a string. This language is regular, with a 2-state DFA that tracks the parity of the number of a's encountered so far.

L_3 equals the set of all strings over $\{a, b\}$! To see this, a string z of length n has $n + 1$-many prefixes, ranging from the empty string to z itself. For $0 \leq i \leq n$, define $f(i) = \#a(x) - \#b(y)$ where $x = z(1) \cdots z(i)$ is the length-i prefix and $y = z(i + 1) \cdots z(n)$ is the rest of z. Then $f(0) \leq 0$, $f(n) \geq 0$, **and** f steps up or down by at most 1 when going from i to $i + 1$, i.e. $|f(i + 1) - f(i)| \leq 1$. Hence there is some i such that $f(i) = 0$ (kind of like the "Intermediate Value Theorem" in calculus), and this i makes $\#a(x) = \#b(y)$ for the corresponding x-y break, so $z \in L_3$. Since z is arbitrary, $L_3 = \Sigma^*$, which is the language of a 1-state DFA, hence regular. (Another way to see this is by induction on strings: $\lambda \in L_3$ with $i = 0$, and for all strings $z \in L_3$ with breakpoint i, $za \in L_3$ with the same breakpoint, while $zb \in L_3$ with breakpoint $i + 1$.)

L_2 is the nonregular language. Take $S = a^*b$, clearly infinite. Let any $x, y \in S$, $x \neq y$ be given. Then there are natural numbers $m \neq n$ such that $x = a^m b$ and $y = a^n b$. Take $z = a^m$. Then $xz = a^m b a^m \in L_2$, but $yz = a^n b a^m \notin L_2$ since there is only one possible "b" breakpoint and it doesn't balance. So $L_2(xz) \neq L_2(yz)$, so the infinite S is PD for L_2, so L_2 is not regular.

A nondeterministic PDA N for L_2 begins with a state q that pushes an A for each a read, but has a choice when a b is read. It can either stay in state q, or it can move to a state r that attempts to pop an A for each a read (ignoring any b read). N accepts if and only if it reads a blank B for empty stack in state r precisely when it also reads the blank on the input tape that marks the end of the input string z. The nondeterminism in state q is needed because there is nothing about any particular 'b' that can tell N whether it will prove to have been a midway one as far as the a's are concerned.