Amortized analysis for two stacks [50 points]

Suppose there are two stacks called $A$ and $B$, manipulated by the following operations:

- **push-A($d$):** Pushes a datum $d$ onto stack $A$. Real Cost = 1.
- **push-B($d$):** Pushes a datum $d$ onto stack $B$. Real Cost = 1.
- **multi-pop-A($k$):** Removes $\min(k, \text{size}(A))$ elements from stack $A$.
  Real Cost = $\min(k, \text{size}(A))$.
- **multi-pop-B($k$):** Removes $\min(k, \text{size}(B))$ elements from stack $B$.
  Real Cost = $\min(k, \text{size}(B))$.
- **transfer($k$):** Repeatedly pops elements from stack $A$ and pushes them onto stack $B$, until either $k$ elements have been moved, or $A$ is empty.
  Real Cost = number of elements moved. (Note that you can transfer only from $A$ to $B$.)

(a) Give amortized costs to each operation using the accounting method. Using your amortized costs show an $O(n)$ worst case bound on the cost of $n$ operations.

(b) Give a potential function such that the amortized cost of each of the operations is constant, and evaluate the constant for each of the operations.

(c) Using your potential function show an $O(n)$ worst case bound on the cost of $n$ operations.
Problem 7-2. Bipartite graphs [50 points]

An undirected graph $G = (V, E)$ is called bipartite if the nodes can be partitioned into two subsets $A$ and $B$ in such a way that all edges go between $A$ and $B$.

(a) Prove that a graph is bipartite iff it can be 2-colored, that is iff all nodes can be colored with two colors so that no two adjacent nodes have the same color.

(b) Give an efficient algorithm to decide whether a graph is bipartite. A by-product of your algorithm should be the partition of $V$ into $A$ and $B$. 