3) (10 pts) DSN (Bitwise Operators)

In the game of NIM, there are several piles with stones and two players alternate taking 1 or more stones from a single pile, until there are no more stones left. The person who takes the last stone wins. It turns out that if it's someone's turn, if they play optimally, they can win as long as the bitwise XOR of all of the number of stones in each pile is not equal to 0. Write a function that takes in an array of values representing the number of stones in the piles of NIM and the length of that array, and returns 1, if the current player can win, and 0 otherwise, assuming both players play optimally.

```
int canWinNIM(int piles[], int numPiles) {

    // Grading: 1 pt to initialize.
    int res = 0;

    // Grading: 3 pts loop, 4 pts XOR
    for (int i=0; i<numPiles; i++)
        res ^= piles[i];

    // Grading: 2 pts, give 1 pt if it says return res, since
    // prompt asks to specifically return 1 in winning case.
    return res != 0;



}
```