# Solutions to Problem Set 5.

5.3-3 Does

```
PERMUTE-WITH-ALL(A)
1. n = A.length
2. for i in 1 to n
3.     swap A[i] with A[RANDOM(1,n)]
```

produce a uniform random permutation?

No. In particular, for $n=3$, there are $3^3$ equally likely sequences of random values generated by the three executions of line 3. The probability of generating a particular permutation $\sigma$ is $k/27$ where $k$ is the number of sequences resulting in $\sigma$. But there are 6 permutations. They can't all result from the same number of sequences because 27 is not a multiple of 6

(It is amusing, but not required, to note that for $n \geq 3$, PERMUTE-WITH-ALL will not produce a uniform random permutation. For $n \geq 3$, $n!$ does not divide $n^n$. In particular, there is a prime factor of $n!$ which does not divide $n$.

If $n$ is prime and $n > 2$, $2 \mid n!$ but $2 \nmid n$.

If $n$ is composite and $n > 2$, there is a prime $p \leq n!$ that does not divide $n$: let $p_1, p_2, \ldots p_k$ be the prime factors of $n$. $2 \leq q = p_1 * p_2 * \cdots p_k - 1 < n$, and $p_i \nmid q$, so $q$ (and $n!$) has a prime factor other than $p_1, p_2, \ldots, p_k$.)

10.1-2 How can you implement 2 stacks in 1 array so that neither overflows until the array is full.

Start one stack at index 1. Increment the top to push and decrement to pop.

Start the other at the last index in the array. Decrement its top to push and increment to pop.

3) Suppose you start with an array of size 1. To insert values into the array, write the values at successive indices. To insert a value into a full array, allocate an array of twice the size, copy the old values into the new array, and write in the new value at the end. Give a $\Theta$-bound on the total number of writes to insert n elements.

| | n | writes to insert $n^{th}$ | cumulative number of writes |
|---|---|---|---|
| $2^0$ | 1 | 1 | 1 |
| $2^1$ | 2 | 2 | 3 |
| | 3 | 3 | 6 |
| $2^2$ | 4 | 1 | 7 |
| | 5 | 5 | 12 |
| | 6 | 1 | 13 |
| | 7 | 1 | 14 |
| $2^3$ | 8 | 1 | 15 |

To insert $2^m$ elements requires $2^{m+1}-1$ writes. ( By induction, $2^{m+1}$ elements require the $2^{m+1}-$ writes to insert the first $2^m$ values, another $2^m$ to copy these into an array of size $2^{m+1}$, then $2^m$ writes to fill the array of size $2^{m+1}$ for a total of

$$2^{m+1} - 1 + 2^m + 2^m = 2^{m+2} - 1, \text{ as required})$$

Let $\omega(n)$ be the number of writes to insert $n$ elements, $n \geq 2$

$$n \leq \omega(n) \leq \omega(2^{\lceil \lg n \rceil})$$

$$n \leq \omega(n) \leq 2^{\lceil \lg n \rceil + 1} - 1$$

$$n \leq \omega(n) \leq 2^{\lg n + 2} - 1 \leq 4(\cancelto{n}{2^{\lg n}})$$

$$n \leq \omega(n) \leq 4n$$

$$\omega(n) = \Theta(n)$$