

Homework Assignment 2

1. Write a version of the SelectionSort algorithm that works in reverse, i.e. it starts building the array from the other end, first correctly placing the maximum element, then correctly placing the element before that etc. The resulting array will still be sorted in increasing order, but it should be “constructed” from right to left as opposed to left-to-right.
2. The following algorithm computes a^n .

```
power(a, n):  
    if n <= 0:  
        return 1  
    else:  
        m <- n div 2  
        b <- a * a  
        if n is even:  
            return power(b, m)  
        else:  
            return power(b, m) * a
```

Using algorithmic analysis, determine the number of multiplications that this algorithm performs as a function of n . You may find it helpful to restrict your analysis to the cases where n is a power of 2.

3. A modern processor can perform around 2×10^9 operations per second. Estimate how much time it would take us to perform a calculation whose number of operations needed is given by the following functions, when $n = 30$ and when $n = 40$.

- a. 2^n
- b. $n!$
- c. n^{30}