Assignment 28

Alex Clemmer
Student number: u0458675

We'll examine the simplest algorithm I could think of. Given $r = a^{b^c} \pmod p$: obviously b^c is c multiplications of b (*e.g.*, $b^3 = b \cdot b \cdot b$). So right there, in our simplest possible algorithm, we have c operations. We take the result of this, let's call it j, and then we perform j multiplications of a. So if $j = 3$, then we multiply a 3 times: $a \cdot a \cdot a$. Our result is is then modded by p, giving us $r = a^j \pmod p$.

The running time should be $O(n^2 \cdot n)$: multiplication of the grade-school variety requires $O(n^2)$ time and we're doing it a variable number (say t) times. Since we know that we do first c multiplications, and then j multiplications, and since we know that $c, j, a < p$, we know that in the worst case this is bounded by the length of p.

If we dealt with arbitrary-length integers, we may be in trouble. Since we're not, it should take $O(n^3)$.