Efficient exponentiation

- Consider the following algorithm:

```
exp(a, b, N) {
    // assume b ≥ 0
    ans = 1;
    for (i=1, i ≤ b; i++)
       ans = [ans * a mod N];
    return ans;
  }
```

- What is the running time?

Efficient exponentiation

- Consider the following algorithm:

```
exp(a, b, N) {
    // assume b ≥ 0
    x=a, t=1;
    while (b>0) {
       if (b odd)
          t = [t * x mod N], b = b-1;
       x = [x^2 mod N],  b = b/2; }
    return t; }
```

- Why does this work?
  - Invariant: answer is [t x^b mod N]
- Running time is polynomial in ‖a‖, ‖b‖, ‖N‖