Test 2

Sam Grayson

April 20, 2015

1. $0 \equiv 3 \cdot 0 \pmod{17}$
   $1 \equiv 3 \cdot 6 \pmod{17}$
   $2 \equiv 3 \cdot 12 \pmod{17}$
   $3 \equiv 3 \cdot 18 \pmod{17}$
   $4 \equiv 3 \cdot 24 \pmod{17}$
   $5 \equiv 3 \cdot 30 \pmod{17}$
   $6 \equiv 3 \cdot 36 \pmod{17}$
   $7 \equiv 3 \cdot 42 \pmod{17}$
   $8 \equiv 3 \cdot 48 \pmod{17}$
   $9 \equiv 3 \cdot 54 \pmod{17}$
   $10 \equiv 3 \cdot 60 \pmod{17}$
   $11 \equiv 3 \cdot 66 \pmod{17}$
   $12 \equiv 3 \cdot 72 \pmod{17}$
   $13 \equiv 3 \cdot 78 \pmod{17}$
   $14 \equiv 3 \cdot 84 \pmod{17}$
   $15 \equiv 3 \cdot 90 \pmod{17}$
   $16 \equiv 3 \cdot 96 \pmod{17}$

   $\{0, 18, 36, 54, 72, 90, 108, 126, 144, 162, 180, 198, 216, 234, 252, 270, 288\}$ forms a complete residue system mod 17. I generated the table above using the following Python code. For an explanation of Python code in general and the source for `linear_diophantine()`, please read 3.23 in my notebook.

```
from tools import linear_diophantine
CRS = []
for n in range(17):
    (x_0, y_0), (r_x, r_y) = linear_diophantine(3, -17, n)
    # now we have $3x_0 - 17y_0 = n$
    # output $n \equiv 3 \cdot x_0 \pmod{17}$
    print (r'${n} \equiv 3 \cdot {x_0} \pmod{{17}}$ \\'.format(**locals()))
    CRS.append(3 * x_0)

# output the whole CRS, seperated by commas
print (', '.join(map(str, CRS)))
```

2. Find $2^{100} \pmod{9}$

   All congruence statements are taken mod 9.

$$
\begin{aligned}
2^{100} &\equiv \quad ? \\
&\equiv 2^{3\cdot 33+1} \\
&\equiv (2^3)^{33} \cdot 2^1 \\
&\equiv 8^{33} \cdot 2 \\
&\equiv (-1)^{33} \cdot 2 \\
&\equiv -1 \cdot 2 \\
&\equiv 7
\end{aligned}
$$

3. **Theorem:**