Problem Source Code

```
from secrets import SystemRandom

gen = SystemRandom()

with open("flag.txt", "rb") as f:
  flag = f.read().strip()

assert len(flag) == 76
c = int.from_bytes(flag)

poly = [c]
for _ in range(10):
  poly.append(gen.randrange(pow(10, 11), pow(10, 12)))

for _ in range(10):
  x = gen.randrange(pow(10, 11), pow(10, 12))
  v = 1
  y = 0
  for i in range(11):
    y += poly[i] * v
    v *= x
  print(f"({x},{y})")
```

Solution

What this code is effectively doing is generating a polynomial of the form $f(x) = c + \sum_{i=1}^{10} a_i x^i$, where $10^{11} \leq a_i < 10^{12}$, and then giving us 10 random points on the polynomial with x-coordinates also in the range $[10^{11}, 10^{12})$. Our goal is to compute c.

We'll do this by computing the coefficients from a_{10} down to a_1 in this order. Let $g(x) = \sum_{i=1}^{10} a'_i x^i$, and we initialize $a'_i = 10^{11}$. Clearly, $f(x) - g(x)$ is a polynomial that is nondecreasing over the positive reals. There exists some minimum value a'_{10} such that $f(x) - g(x)$ eventually decreases, so we can binary search for this value and set a'_{10} to the largest integer less than this value. We repeat this process for a'_9 going down to a'_1. Once we're done, we're left with the constant term and we can print it out directly.

Source code can be found here.

Notes on Correctness

Imagine that we guess $a'_{10} = a_{10} + 1$. The derivative of $f(x) - g(x)$ can be approximated by $-10x^9 + 9a_9x^8$, which becomes negative when $x = \frac{9a_9}{10}$. The given source code could fail if all x-values happened to be less than this value. However, because the polynomial's first through ninth derivatives are all nonnegative, we could take finite differences to validate that all of the underlying sequences were still positive on the range, and if any of these checks failed, then the guessed value in our binary search would still be too large.

For the lower-order terms, the derivative becomes negative at increasingly smaller values, so if we guess the value of a_{10} correctly, the rest of the terms should be correctly guessed as well.