Spring 2019, Math 596: Problem Set 3
Due: Tuesday, February 19th, 2019
Factorization Lengths and Sage

Sage lab discussion. The lab exercises below should be completed in assigned pairs (though each at your own machine). Be sure to run each of the commands listed, but don't be afraid to go on tangents and try some different/related things. One of the best ways to become familiar with new software is to dive in and get your hands dirty!

(L1) *Numerical semigroups package in Sage.*

The `NumericalSemigroup.sage` package contains a large collection of commands that will be useful throughout our research this summer. The package can be loaded with the following command.

```
load('/PATH/TO/PACKAGE/NumericalSemigroup.sage')
```

The main object we will use from this package is the `NumericalSemigroup` object, which can be created as follows.

```
S = NumericalSemigroup([6,9,20])
S
```

The `NumericalSemigroup()` function takes one parameter, namely a list of generators. In this case, this list is `[6,9,20]`, making `S` the McNugget semigroup. Note that the following will result in an error, since the `NumericalSemigroup()` function accepts a single argument.

```
S = NumericalSemigroup(6,9,20)   # THIS WILL ERROR
```

The NumericalSemigroup object will automatically prune the list of generators down to the minimal generating set.

```
S = NumericalSemigroup([6,9,15,20,26,44])
S.gens
```

Tab autocomplete is an extremely useful feature. To activate, type "`S.`" and then press the tab key, and a list of available functions will appear. Typing "`S.Fa`" and then pressing tab will show all available functions beginning with `Fa`.

(L2) *Computing various quantities in Sage.*

The `NumericalSemigroup.sage` package can compute most of the quantities we saw this morning. Let's start by creating a semigroup.

```
S = NumericalSemigroup([6,9,20])
```

Sets of factorizations can be computed using the `S.Factorizations()` function. Try running this with some different semigroup elements (keep in mind that larger elements will require more time to compute!)

```
print S.Factorizations(20)
print S.Factorizations(200)
print S.Factorizations(2000)
```

The output from the last one is a bit much to handle on its own. Perhaps we are only interested in how *many* factorizations there are:

```
len(S.Factorizations(2000))
```

Or, perhaps we are only interested in the factorization *lengths* that can occur.

```
S.LengthSet(2000)
```

Let's verify the Apéry set of the McNugget semigroup.

```
S.AperySet(6)
```

Notice the elements are listed based on their value modulo 6, and not in increasing order. Try computing the Apéry set with respect to some other elements.

```
S.AperySet(100)
```

Use tab autocomplete to try running a few other random functions on `S`. Be adventurous! It is (hopefully) hard to break things.

(L3) *Plots, plots, and more plots.*

Let's mess around with some of Sage's plotting capabilities. The most commonly used function in past projects is `points()`, which plots a list of points.

```
points([(1,1), (1,2), (2,1), (2,2)])
```

Use tab autocomplete to see the list of options for the `points()` function. Plots can be overlayed by simply "adding them together" as in the following code.

```
plot1 = points([(1,1), (1,2), (2,1), (2,2)])
plot2 = points([(1,1), (2,2), (3,3)], color="red", size=30)
plot1 + plot2
```

One resounding theme: well-chosen plots can be surprisingly illuminating. Sometimes, creating the "right" plot can highlight exactly what you are looking for. To demonstrate this, let's create a plot of the max factorization length function for `S`.

```
pts = []
for i in [1..200]:
    if i in S:
        val = max(S.LengthSet(i))
        pts.append((i,val))
    else:
        pts.append((i,0))
points(pts)
```

How can we tell from this plot that maximum factorization length is eventually quasilinear?

Sage discussion problems. The problems below should be worked on in the computer lab.

(D1) *Warmup.* Let $S = \langle 33, 57, 91, 105 \rangle$. Use Sage to do the following.

(a) Find $\mathsf{Z}(1000)$ and $\mathsf{L}(1000)$.

(b) Find $\mathrm{Ap}(S; 33)$ and $\mathrm{Ap}(S; 105)$.

(c) Find $\mathsf{PF}(S)$ and $\mathsf{t}(S)$.

(d) Find an irreducible numerical semigroup T containing S with $\mathsf{F}(T) = \mathsf{F}(S)$. Find $\mathsf{e}(T)$.

(D2) *Eventually quasipolynomial functions.* Let $S = \langle n_1, \ldots, n_k \rangle$ with $n_1 < \cdots < n_k$.

(a) Write Sage code that creates a plot with a point at (n, d) whenver $n \in S$ and $d \in \Delta(n)$. Note that there may be several points with identical first coordinate. What do you notice for large n?

(b) Let $S = \langle n_1, \ldots, n_k \rangle$ with $n_1 < \cdots < n_k$. For $n \in S$ and $(a_1, \ldots, a_k) \in \mathsf{Z}(n)$, define

$$\|(a_1, \ldots, a_k)\|_\infty = \max(a_1, \ldots, a_k),$$

called the ℓ_∞-norm of $(a_1, \ldots, a_k)$. Let $M(n)$ and $m(n)$ denote the maximum and minimum ℓ_∞ norms of factorizations of n, respectively. It turns out $M(n)$ and $m(n)$ are both eventually quasilinear (just like maximum and minimum factorization length). Determine their periods and leading coefficients in terms of $n_1, \ldots, n_k$.

(c) Write Sage code that computes the smallest positive integer N so that $n \mapsto \max \mathsf{L}(n)$ is quasilinear for all $n \geq N$ (that is, so that $\max \mathsf{L}(n + n_1) \neq \max \mathsf{L}(n) + 1$). The value of N is called the *dissonance point*.

Homework problems. You are required to submit all of the problems below. For those with choices, you may submit additional parts if you wish, but please indicate what you want graded. Although I am happy to provide written feedback on all submitted work, no extra credit will be awarded for completing additional problems.

(H1) Locate a numerical semigroup S with an Apéry set element $n \in \mathrm{Ap}(S)$ such that $|\mathsf{L}(n)| \geq 2$.

(H2) Locate a numerical semigroup S with an element $n \in S$ satisfying $|\mathsf{L}(n)| = 1$ and $|\mathsf{Z}(n)| = 2$.

(H3) Complete *at least one* of the following.

(a) Let $S = \langle n_1, \ldots, n_k \rangle$ with $n_1 < \cdots < n_k$. Let $\mathsf{m}(n)$ denote the minimum factorization length of n. Prove that for $n > n_k^2$, we have

$$\mathsf{m}(n + n_k) = \mathsf{m}(n) + 1.$$

Hint 1: Lemma 4.1 in the paper *On the set of elasticities in numerical monoids* from my webpage may be useful in your proof here.

Hint 2: a full proof for maximum factorization length can be found in Theorem 4.2 of the aforementioned paper.

(b) Prove that

$$\min \Delta(S) = \gcd(\{n_i - n_{i-i} : 2 \leq i \leq k\})$$

for any numerical semigroup $S = \langle n_1, \ldots, n_k \rangle$.

Challenge problems. Challenge problems are not required for submission, but bonus points will be awarded for submitting a partial attempt or a complete solution.

(C1) Prove that the functions $M(n)$ and $m(n)$ from (D2)(b) are eventually quasilinear.