A Penchick's Tale

> **Problem Solving Technique: Why is this hard?**
> Explore straightforward "just do it" approaches or thought processes, even if they seem wrong. Articulate precisely *what's stopping you*, i.e. what is the heart of what makes the problem hard—this gives you something concrete to try to attack.

Can't I just make the maximum amount of each of `squawk` and `chirp` and `quack`, as much as possible? Making more is always better, right?

Unfortunately, as stated, this solution is ill-defined, because `quack` conflicts with the letters in `squawk` and in `chirp`.

So, what makes the problem hard?

- The number of `quack` we construct affects the number of `squawk` and `chirp` we can construct, which makes things messy.
- The problem is hard because there is a tradeoff made whenever we make a `quack` instead of a `squawk` and `chirp`.
- We can't just always max out on one option—we need to find "the right number" of `quack` to make.
- How to compute this magic number is not obvious (and depends on the values of a and b and c)—it feels hard to express in a formula.

But while thinking about conflicts, we might notice that...

- If I'm only considering `squawk` and `chirp`, their letters don't conflict.
- So, considering just them, I can greedily make as much of each of them, since they don't interfere with each other.

The problem has now been reduced to this key goal:

- Find the correct number of `quack` to make.

We have a standard tool for that.

> **Standard Toolbox: Complete Search**
> Can't figure out the right value? Just try everything.
> - **Pros:** Always works
> - **Cons:** Might be too slow (do the analysis)

We now have the following solution sketch.

For each z from 0 to (max no. of `quack` that can be made):

- Make z copies of `quack`

- Using the remaining tiles, makes as many `squawk` and `chirp` as possible; let these values be x and y.

- The maximum happiness in this case is $ax + by + cz$.

The answer to the problem is the maximum value of $ax + by + cz$ across all possible values of z.

The maximum no. of `quack` that can be made is definitely $< |s|$. Each iteration of the loop can be done in $O(1)$.

Thus, this solution runs in $O(|s|)$, which gets 100 points.

> **Remark**
> There are many other greedy solutions to this problem, but it's trickier to prove their correctness.