1 Part1

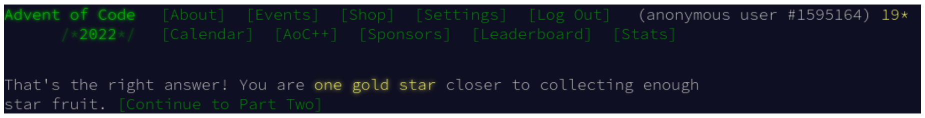
Solving such problems in LaTeXsounds like a dump idea. And this is totally true. So it is possible to write something like code in LaTeXbut the real problem is that all you do is defining macros. So it took me a while but I came up with a solution for Part 1. Parsing input files is not that difficult. The first problme that occured where nested loops. Somehow if I am done with the inner one I also leave the outer one. So some nice people on stackexchange had the same problem and the solution provided was what you can see in 1.

```
\loop\unless\ifeof\file%
    \read\file to\input%
    \setsepchar{ }%
    \readlist*\foo{\input}%
    \setcounter{cnt}{1}%
    \foreachitem\z\in\foo
    {%
        \ifnum\zcnt=1%
            \operation{\foo[\zcnt]}
        \fi%
        \ifnum\zcnt=2%
            \setNumber{\foo[\zcnt]}
            \pgfmathsetmacro{\points}{\fpeval{\nr+\points}}
        \fi%
    }%
\repeat%
```

Listing 1: nested loops

As you can see the inner curly brackets solved that problem. So you don't need them, or at least only need curly brackets around loop from time to time. It depends.

Afterwards I needed to calculate some stuff. LaTeXis also not made for doing this. But luckily there are some packages like intcalc which are able to calculate. And putting all of it together I came up with the following number as solution for part 1: 13760And guess what after submitting 13760I saw the following:


Advent of Code [About] [Events] [Shop] [Settings] [Log Out] (anonymous user #1595164) 19*
/*2022*/ [Calendar] [AoC++] [Sponsors] [Leaderboard] [Stats]

That's the right answer! You are one gold star closer to collecting enough star fruit. [Continue to Part Two]

Figure 1: a nice plot

So I onto part 2 I guess.

2 Part 2

Not so easy as I hoped. Or at least not so easy to solve in LaTeX. The first problem and also the problem which leaded me to give up on this and do it in python was the following. There is nothing like if a and b. And nothing like $a \leq b$.
Well python also is a nice language. And in compared to the LaTeXstuff in about no time I came up with a solution. All the magic behind are those 13 lines of code.

```
def do_cycle():
global crt
global pixels

if (cycle - 20) % 40 == 0:
    signal_strengths.append(cycle * X)

pixels += "#" if X - 1 <= crt <= X + 1 else "."
crt += 1 if crt < 39 else -39

if crt == 0:
    pixels += "\n"

return 1
```

Listing 2: doing python

Listing 2 lead to the following.

```
###..####.#..#.####..##..###..####.####.
#..#.#....#.#.....#.#..#.#..#.#....#....
#..#.###..##.....#..#....#..#.###..###..
###..#....#.#...#...#....###..#....#....
#.#..#....#.#..#....#..#.#....#....#....
#..#.#....#..#.####..##..#....####.#....
```

Listing 3: help me to read

Idk it took me like 5 minutes to extract all the letters out of this stuff.