

# README

Before you get famous by solving the infamous zodiac cipher you may read this short documentation

## What is this repository for?

This repository contains my work on the z340 cipher. This project contains lots of code to analyze and manipulate the cipher and I have created many test cases. I tried to write a convenient library which allows to transpose the cipher as easy as possible and I hope you find it useful. It would be great if you add your own ideas!

There is no solver included in this project. Pass your generated results to AZDecrypt or zkDecrypt.

## How do I get set up?

### Prerequisites

The project is written in Python 2.7.10 since Python is a very powerful and convenient language. If you are using windows you have to install a python interpreter before installing the IDE. Make sure to download the newest **2.7.x** release, not **3.x** !!!

<https://www.python.org/downloads/>

I use the PyCharm Community Edition IDE which is available for free (Windows, macOS :

<https://www.jetbrains.com/pycharm/download/>

Personally I use macOS and Windows. Since most of the users may use Windows as their primary os I will describe the setup process for Windows. These steps should be very similar when you are using Mac OS or Linux.

## Setup PyCharm Community Edition and run the project

- Just run the setup file and follow the installation steps. I recommend the option „Darcula“ for the settings „IDE theme“ and „Editor colors and fonts“.
- Click „Open Project“ and select the folder of the cloned git repository.
- Click „File -> Settings -> Project ‚Zodiac Investigator‘ -> Project Interpreter“ and select the previously installed python interpreter.
- Click again „File -> Settings -> Project ‚Zodiac Investigator‘ -> Project Interpreter“ and then the small green „+“ next to „Packages“. Now search a package named „Pillow“ and install it.
- Now right-click on „main.py“ and choose „Run ‚main‘“ from the context menu. The project should run now without errors.....and without any results since I have commented out all the tests. Feel free to comment-in tests and analytic functions in main.py and check the results!
- Have fun!

# Project Structure

The following section describes the project structure.

## Root-Directory:

Here you can find all python files. These are:

### **main.py**

Call all your test- and analytic functions from here. Do not implement new tests or analytic functions here.

### **manipulation.py**

Implement all your transpositioning ideas here!

### **analyzing.py**

Implement all your analyzing stuff here!

### **globals.py and helpers.py**

Guess!

### **snippet.py**

A class which represents a cipher. This class includes a lot of functions to manipulate the cipher (rotating, resizing, picking parts/rows/columns and put them back...)

### **investigator.py**

A class for processing outputs, applying keys, create plus patterns.

### **chromosome.py, population.py and population settings.py,**

Classes for genetic algorithms. Experimental stuff by now but working

## Directory „Testcases“

Put all your test ciphers here

## Directory „Transcriptions“

You can find z340, z408, z32 and some other transcriptions here

## Directory „Skipping Letter Patterns“

I wrote a test which removes letters from a cipher by using images which contain a pixel pattern. This directory contains some test patterns.

Check function `create_skip_letters_pixel_pattern` in file `manipulation.py` for more information

## Directory „Generated Patterns“

All the output which is generated by the manipulation functions can be found here. Subfolders are generated from the list in „investigator.py“

## Directory „Plus Patterns“

I think that the high amount of + symbols in z340 are the result of some shuffled pattern caused by transposition. If you send your test results to the investigator you can choose if a plus pattern should be created and analyzed. The results can be found here. Subfolders are the same like in „Generated Patterns“

## Contribution guidelines

If you want to create your own tests please create a branch! I will merge your branches into the master branch.

Please make sure that your coding style matches the code in this project. I will not write coding guidelines so please use the existing code as a reference.