

Author: Andreas Traut

Date: 22.02.2021

[Download as PDF](#)

## Algorithms, Data Structures and Coding

### 0. Introduction

- a) Aim of this repository
- b) Motivation for this repository
- c) Structure of this repository
  - (i) First part: How to improve your coding skills: Certificates and Challenges
  - (ii) Second part: Examples

### I. How to improve your coding skills: Certificates and Challenges

- 1. Earn a certificate
- 2. Get into coding challenges

### II. Examples

- 1. Python-Examples
- 2. Excel Example
- 3. Access-Example

MIT License

# Algorithms, Data Structures and Coding

---

## 0. Introduction

---

### a) Aim of this repository

The aim of this repository is to improve coding skills and enhance skills in algorithmic thinking and data structures.

### b) Motivation for this repository

I am programming in different languages and environments for nearly my whole life:

- Starting in the 1980s / 1990s with [GW-Basic](#) and the integrated development environment (IDE) [Turbo Pascal](#).
- Then in the 2000s / 2010s I started with [C++](#), where I understood the **object oriented way of thinking**. I learnt a lot in C++ at my final years at university as well as during the first years at my first employer. Today my 8 cm thick "*C++ programming bible*" serves as an elevation for my second monitor, which I had to set up due to the Corona-related home office.
- I used [SQL](#) a lot and also got quite skilled in finding solutions with [Visual Basic \(VBA\)](#). VBA (applied in Excel or [Access](#)) is fun for me and served me a lot during my whole professional and private.
- In 2019 I learnt the advantages of the [Jupyter-Notebooks](#): beautiful, intuitive, easy to use and build. But there is something, I don't like in Jupyter-Notebooks, which I will explain below.
- And today I am a big fan of [Python](#): it's so much more fun to use Python instead of C++: I enjoyed not having these opening brackets `{` and closing brackets `}` and `;` at the end of a line! Such a relieve for my eyes.

I am glad, that lots is similar in all these decades: **the way of thinking as a programmer**. My motivation is to give you some basic hints, advises and guidance to improve your coding skills.

## c) Structure of this repository

### (i) First part: How to improve your coding skills: Certificates and Challenges

In the *first part* I will explain, how certificates and coding challenges can be useful for you to improve your coding skills.

### (ii) Second part: Examples

In the *second part* I will work on some interesting examples.

## I. How to improve your coding skills: Certificates and Challenges

---

### 1. Earn a certificate

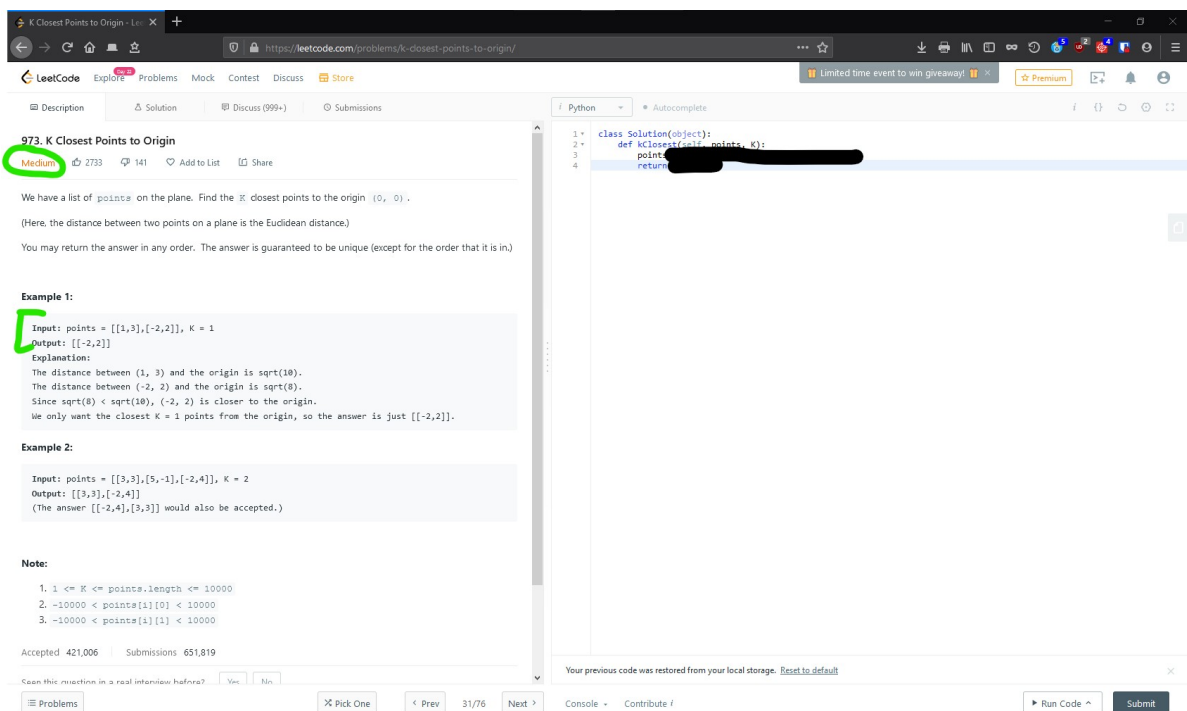
A good way for improving your coding skills are by going through some online courses and trying to earn a certificate. There are a lot of other resources: maybe start getting an overview on [Coursera](#). These courses are nice because the teachers are usually highly skilled (from universities) and the technical infrastructure for the courses is rather advanced: there are videos with subtitles and transcript and you can easily navigate through these videos by reading across these transcripts and jumping to the positions in the video, which you want to listen to. You can monitor your learning curve and weekly progress. But the Coursera certificates usually cost some money.

If you want to find something cheaper, then I can recommend the ["Data Structures and Algorithms in Python"](#) from Jovian. When I worked for it in 02/2021 it was for free. It uses Jupyter-Notebooks and is definitely a lot of fun! You will learn in video tutorials and practise with well documented Jupyter-Notebooks how to work with python classes, binary trees, sorting algorithms and understand how to solve coding problems systematically.

There are various other resources for earning a certificate and listing up, what I found is not very helpful at the end for you: try to find **the certificate which YOU want to earn!** I promise: working for it is a lot of fun.

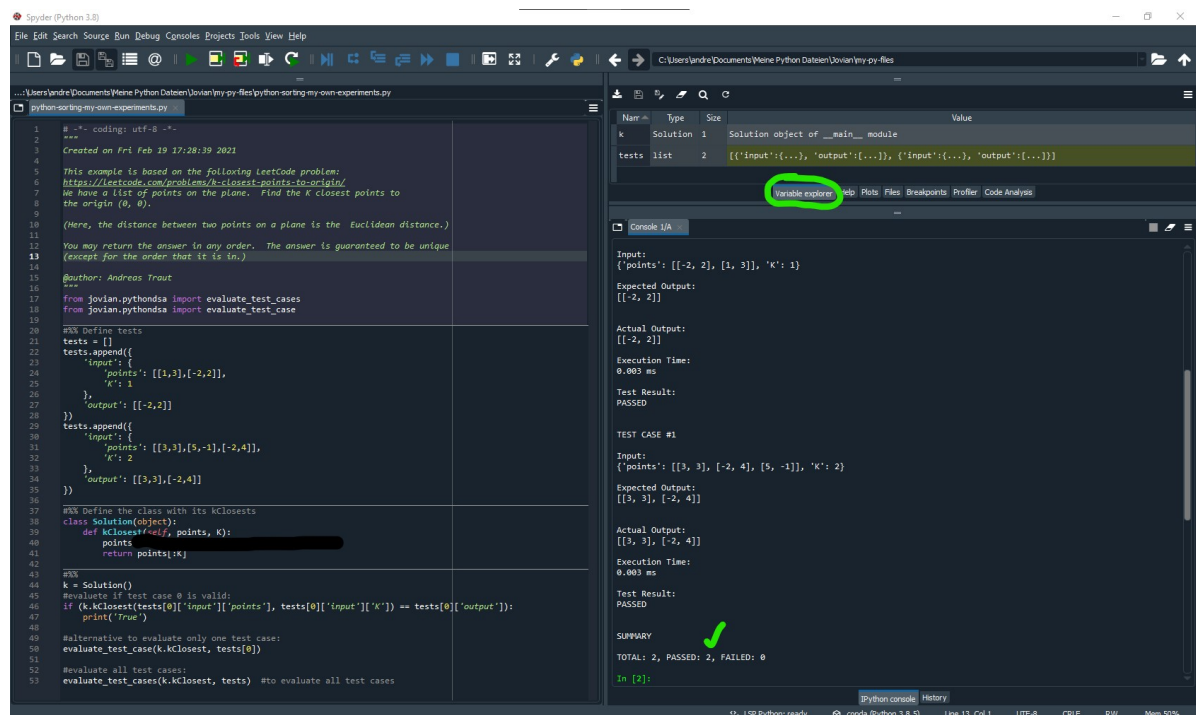
### 2. Get into coding challenges

Another advice I can give you is to get into coding challenges. When you accept a coding challenge, then a problem will be shown and would have to solve it in your preferred programming language (python, java, C++,...). I tried [LeetCode](#) and you will find a lot of other websites, which provide similar concepts. On the left is the problem, on the right some place to program a solution:



Some of these problems were quite interesting for me so I wanted to have them in my integrated development environment (IDE) [Spyder-IDE](#) in order to debug through the code and extend the examples a bit. I recommend to do use an integrated development environment (IDE) as often as you can, instead of always going through Jupyter Notebooks. In my opinion Jupyter Notebooks are **not** always the best environment for learning to code! I agree, that Jupyter Notebooks are nice for doing documentation of python code. It really looks beautiful. But I prefer debugging in an IDE instead of a Jupyter Notebook: having the possibility to set a breakpoint can be a pleasure for my nerves, specially if you have longer programmes. Some of my longer Jupyter Notebooks feel from the hundreds line of code onwards more like pain than like anything helpful. And I also prefer having a "help window" or a "variable explorer", which is smoothly integrated into the IDE user interface. And there are a lot more advantages why getting familiar with an IDE is a big advantage compared to the very popular Jupyter Notebooks! I am very surprised, that everyone is talking about Jupyter Notebooks but IDEs are only mentioned very seldomly. But maybe my preferences are also a bit different, because I grew up in a [MS-DOS](#) environment. :-)

Here is how the problem from above looks like in the Spyder-IDE:



Another example from LeetCode: the [Sort-Colors](#) problem:

The image shows two screenshots. The top screenshot is of the LeetCode 'Sort Colors' problem page. The problem description states: 'Given an array `nums` with `n` objects colored red, white, or blue, sort them in-place so that objects of the same color are adjacent, with the colors in the order red, white, and blue. We will use the integers `0`, `1`, and `2` to represent the color red, white, and blue, respectively.'

Example 1:  
Input: `nums = [2,0,2,1,1,0]`  
Output: `[0,0,1,1,2,2]`

Example 2:  
Input: `nums = [2,0,1]`  
Output: `[0,1,2]`

Example 3:  
Input: `nums = [0]`  
Output: `[0]`

Example 4:  
Input: `nums = [1]`  
Output: `[1]`

Constraints:

- `n == nums.length`
- `1 <= n <= 300`
- `nums[i]` is `0`, `1`, or `2`.

The bottom screenshot shows the Spyder Python IDE. The editor displays a Python solution for the 'Sort Colors' problem. The solution defines a `Solution` class with a `sortColors` method. The method uses a counting sort approach: it creates a copy of the input list, counts the occurrences of each color (0, 1, 2), and then reconstructs the sorted list by iterating over the counts. The console output shows the execution of the solution for the example inputs, confirming the expected outputs.

## II. Examples

In the *second part* I will work on some interesting examples, which will be available as `.py` Python-Files or will be Tools like Excel/VBA or Access.

### 1. Python-Examples

I already provided a lot of Jupyter-Notebooks in Python but will provide further `.py` Python-Files here in short time.

## 2. Excel Example

During my career I implemented a lot of Excel/VBA solutions: one was a Excel/VBA project management tool, which organized and structured a complex project flow of a team of 7 people. My Excel/VBA solution is used on a daily basis and is running for already 2 years now.

I won't be able to mention all the other Excel/VBA which I built or worked on. But I will provide an example of an Excel/VBA solution in short time here.

## 3. Access-Example

During my career I also worked with Access solutions. One was for a team of 20 people who were working simultaneously with their Access-frontends on one Access-backend. The aim was to assure a structured data-entry of the whole team into the Access backend database by using Access-Forms. At the end of the project the filled database tables had been joined with SQL queries to further databases in order to aggregate a very specific result table.

I also worked on other Access solutions and will provide an example of an Access solution in short time here.

---

# MIT License

---

MIT License  
<https://opensource.org/licenses/mit-license.php>

Copyright (c) 2021 Andras Traut

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.