# Update a File Through a Python Algorithm

## Project Description

At my organization, access to restricted content is controlled with an allow list of IP addresses. The 'allow\_list.txt' file identifies these IP addresses. A separate remove list identifies IP addresses that should no longer have access to this content. I created an algorithm to automate updating the 'allow\_list.txt' file and remove these IP addresses that should no longer have access.

## Open the File That Contains the Allow List

I used the `with` statement and `open()` function in read mode to open the allow list file. This method ensures proper resource management by closing the file automatically. Example: `with open(import\_file, 'r') as file:`

## Read the File Contents

To read the file contents, I used the `.read()` method and stored the string output in the variable `ip\_addresses`. This method reads the entire file content into a single string.

## Convert the String into a List

The `.split()` method was used to convert the `ip\_addresses` string into a list. This allows easy removal of individual IP addresses.

## Iterate Through the Remove List

I created a `for` loop to iterate over the `remove\_list`. Each IP address in the `remove\_list` was checked against `ip\_addresses`.

## Remove IP Addresses That Are on the Remove List

Inside the loop, I used a conditional statement to check if the IP exists in `ip\_addresses`. If it does, I used the `.remove()` method to delete it. This method works here because there are no duplicates in the list.

## Update the File with the Revised List of IP Addresses

After the list was updated, I used the `.join()` method to convert it back into a string, separated by new lines (`'\n'`). Then, I used another `with` statement and the `.write()` method to overwrite the file with the updated list.

## Summary

This algorithm automates the update of an IP allow list. It includes opening and reading a file, converting its contents to a list, removing specified elements, and writing the updated data back to the file. This technique ensures secure, up-to-date access control in environments handling sensitive information.