# Algorithm for file updates in Python

## Project description

Create an algorithm to check whether the allow list file contains any IP addresses identified on the remove list. If so, you should remove those IP addresses from the file containing the allow list. This is eventually put into a function and could be used to remove any elements from the file.

## Open the file that contains the allow list

```
import_file = "allow_list.txt"
with open(import_file, "r") as file:
```

We use `with` to automatically destroy the file object once we exit the block.

## Read the file contents

```
ip_addresses = file.read()
```

## Convert the string into a list

```
ip_addresses = ip_addresses.split()
```

The `split()` function splits a string into a list on a specified delimiter, the default with no argument is whitespace, so space or \n.

## Iterate through the remove list

```
for element in ip_addresses:
```

## Remove IP addresses that are on the remove list

```
    if element in remove_list:
        # then current element should be removed from `ip_addresses`
        ip_addresses.remove(element)
```

`list.remove(element)` allows us to remove any element from a list. We use a loop to go through all possible elements to be removed and remove each one individually.

## Update the file with the revised list of IP addresses

```
ip_addresses = " ".join(ip_addresses)

with open(import_file, "w") as file:

  # Rewrite the file, replacing its contents with `ip_addresses`

  file.write(ip_addresses)
```

## Summary

This lab exercise went through the process of parsing and editing files in Python. This task could be done occasionally by cybersecurity analysts and is a great introduction to the topic of using Python for cybersecurity automation.