

## Solution Review: Remove Even Integers from a List

This review provides a detailed analysis of the different ways to remove even integers from a list.

We'll cover the following



- Solution 1: Doing it “by hand”
  - Time Complexity
- Solution #2: Using list comprehension
  - Time Complexity

### Solution 1: Doing it “by hand” #

```
1 def remove_even(list):
2     odds = [] # Create a new empty list
3     for number in list: # Iterate over input list
4         # Check if the item in the list is NOT even
5         # ('%' is the modulus symbol!)
6         if number % 2 != 0:
7             odds.append(number) # If it isn't even append it to the empty list
8     return odds # Return the new list
9
10
11 print(remove_even([3, 2, 41, 3, 34]))
12
```



This solution starts with the first element of the list and checks if it is not even. If it is odd, the element is appended to a new list. Otherwise, it skips to the next element. This repeats until the end of the list is reached.

You might have written a solution like this one. It isn't *wrong*, it's not very Pythonic. Python is known for its economical code so we'll be introducing ways to make your solutions as Pythonic as possible throughout this course!

### Time Complexity #

Since the entire list has to be iterated over, this solution is in  $O(n)$  time.

### Solution #2: Using list comprehension #

```
1 def remove_even(list):
2     # List comprehension to iterate over List and add to new list if not even
3     return [number for number in list if number % 2 != 0]
4
5
6 print(remove_even([3, 2, 41, 3, 34]))
```



A Python technique called **list comprehension** is used to iterate over the initial array. With list comprehension, checking a condition and appending to the new list can all be done in one line. The code for it starts and ends with a '[' and ends with a ']'. The basic syntax is:

```
newList = [expression(i) for i in oldList if filter(i)]
```

The list is iterated. If the number is odd, it is appended to a list to be returned, and if even the element is filtered out from the list. Repeat until the end of the list is reached.

## Time Complexity #

The time complexity of this solution is also  $O(n)$ , since only the syntax has changed while the algorithm still iterates over all elements of the list.

[← Back](#)[Next →](#)

Challenge 1: Remove Even Integers fr...

Challenge 2: Merge Two Sorted Lists

 **Completed**



Report an  
Issue



Ask a Question

([https://discuss.educative.io/tag/solution-review-remove-even-integers-from-a-list\\_\\_introduction-to-lists\\_\\_data-structures-for-coding-interviews-in-python](https://discuss.educative.io/tag/solution-review-remove-even-integers-from-a-list__introduction-to-lists__data-structures-for-coding-interviews-in-python))