Interview Cake

← course home (/table-of-contents)

Array Slicing

Array slicing involves taking a subset from an array and **allocating a new array with those elements**.

In Python 3.6 you can create a new list of the elements in `my_list`, from `start_index` to `end_index` (exclusive), like this:

Python 2.7

```
my_list[start_index:end_index]
```

You can also get everything from `start_index` onwards by just omitting `end_index`:

Python 2.7

```
my_list[start_index:]
```

Careful: there's a hidden time and space cost here! It's tempting to think of slicing as just "getting elements," but in reality you are:

1. allocating a new list
2. *copying* the elements from the original list to the new list

This takes $O(n)$ time and $O(n)$ space, where n is the number of elements in the *resulting* list.

That's a bit easier to see when you save the result of the slice to a variable:

Python 2.7

```
tail_of_list = my_list[1:]
```

But a bit harder to see when you don't save the result of the slice to a variable:

Python 2.7

```
return my_list[1:]
# Whoops, I just spent O(n) time and space!
```

Python 2.7

```
for item in my_list[1:]:
    # Whoops, I just spent O(n) time and space!
    pass
```

So keep an eye out. Slice wisely.