In [32]:
```python
# Example 26: np.max(), np.min(), np.argmax(), np.argmin()

a = np.array([0,1,2,3,16,3,2,1,0])
print('a:',a)
print('max of a =',np.max(a))
print('min of a =',np.min(a))
print('index of max value of a =',np.argmax(a))
print('index of min value of a =',np.argmin(a))
```

```
a: [ 0  1  2  3 16  3  2  1  0]
max of a = 16
min of a = 0
index of max value of a = 4
index of min value of a = 0
```

Questions

These questions are in no particular order (except for question 0, do that one first). The questions range in difficulty; some are one-liners, others require a lot more thinking. Don't be discouraged if you hit a roadblock. Talk to your neighbors and ask for help from the lab staff.

Question 0

In order to test your code, plase run the cell below to load the autograder. There is a cell after each question that you can run in order to check your answer. The autograder is purposefully not very verbose.

In [33]:
```python
%run autograder.py
```

```
Autograder loaded!
Question 0 Passed!
```

Question 1

Search the NumPy documentation and/or the web for a NumPy function that can solve a system of linear equations of the form `Ax=b`. Once you've found the package and function, insert into the `func` placeholder below. Often Googling `Numpy` followed by the function you'd like will give you exactly what you want. Ex: Try searching `Numpy Matrix Solver`

In [40]:
```python
# find the missing package/function
func = np.linalg.solve # YOUR CODE HERE

# Do not modify the code below
def q1(A,b):
    return func(A,b)
```

In [41]:
```python
test_q1(q1)
```

```
Question 1 Passed!
```