

**Find duplicate numbers in a given list a**

- 1. List a, of size of n, given**
- 2. Each list element is between 0 to (n- 1)**


<b>n = 4</b>
<b>0 1 2 3</b>
<b>3 2 1 0</b>
<b>No dup</b>

<b>n = 4</b>
<b>0 1 2 3</b>
<b>2 3 1 3</b>
<b>duplicate {3}</b>

<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>
<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>3</b>	<b>6</b>	<b>6</b>
<b>duplicate {1, 3, 6}</b>						

**Goal:**

- 1. Collect all duplicates exactly once in the list d**

**wrong** 

<b>0</b>	<b>1</b>	<b>2</b>
<b>1</b>	<b>1</b>	<b>1</b>
<b>d = {1, 1}</b>		

<b>0</b>	<b>1</b>	<b>2</b>
<b>1</b>	<b>1</b>	<b>1</b>
<b>d = {1}</b>		

- 2. Given list a should be intact at the end of your algorithm**

- 3. We need O(n) time and O(1) space algorithm**

**Implement O(n) time and O(1) space algorithm**

**You cannot use any library functions like set/map etc.  
Must use only Python list []**