1 Data Structures

1.1 Linked List

- Motivation: implementation of list using array needs to occupy contiguous memory space (can result in memory error)
- Variants of linked list:
  - Tailed (need to maintain `head` and `tail`)
  - Circular
  - Doubly linked (`prev` and `next` attributes for `ListNode`)
- How to find cycle?
  Answer: use fast and slow pointers

```
    slow = slow.next;
    fast = fast.next.next;
```

- **[IMPT]** Drawing pictures is very important to visualize the program!

Java API: `ArrayList` or `LinkedList`

```
\\ constructor
ArrayList<Integer> list = new
 ArrayList<Integer>;
```

2 Algorithms

2.1 Sorting

3 Java Tricks

- Use `StringBuilder` for return statements
  - Java StringBuilder API
  - Zigzag conversion