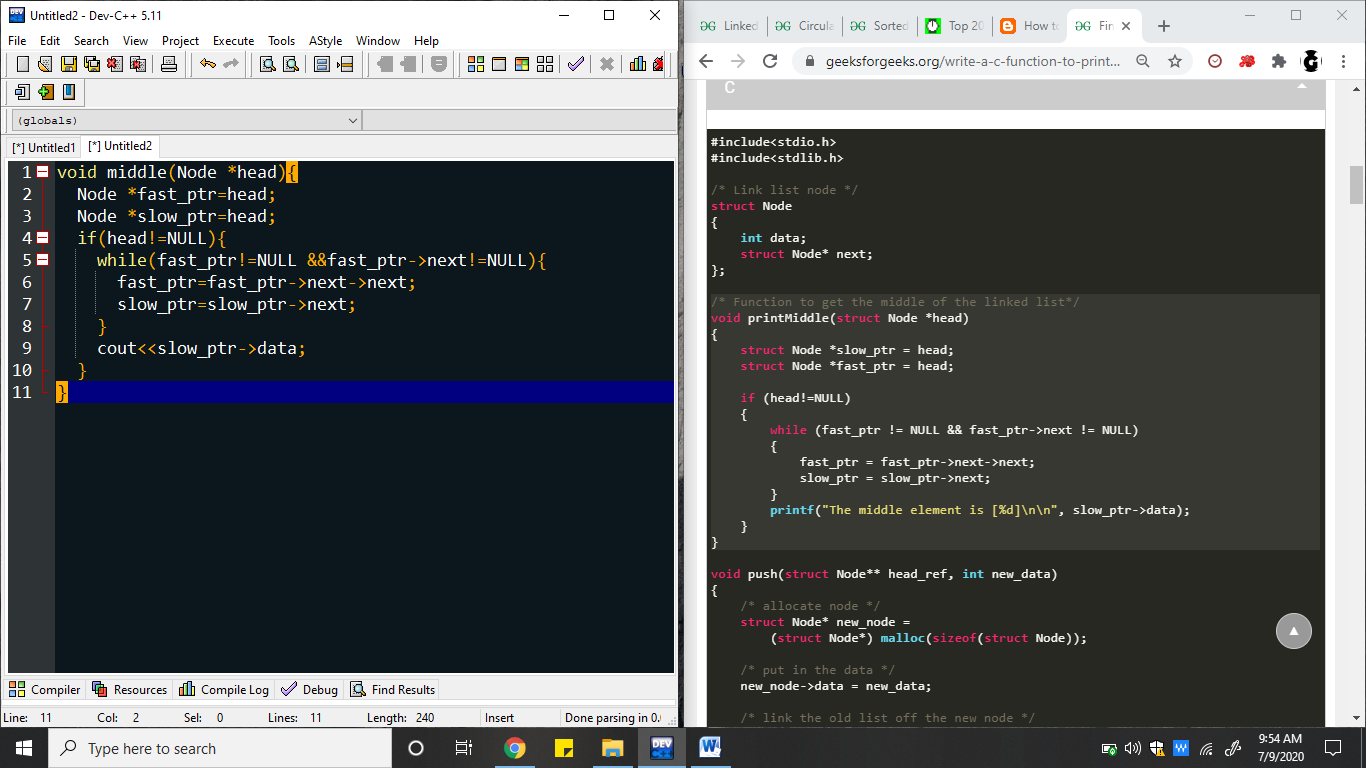
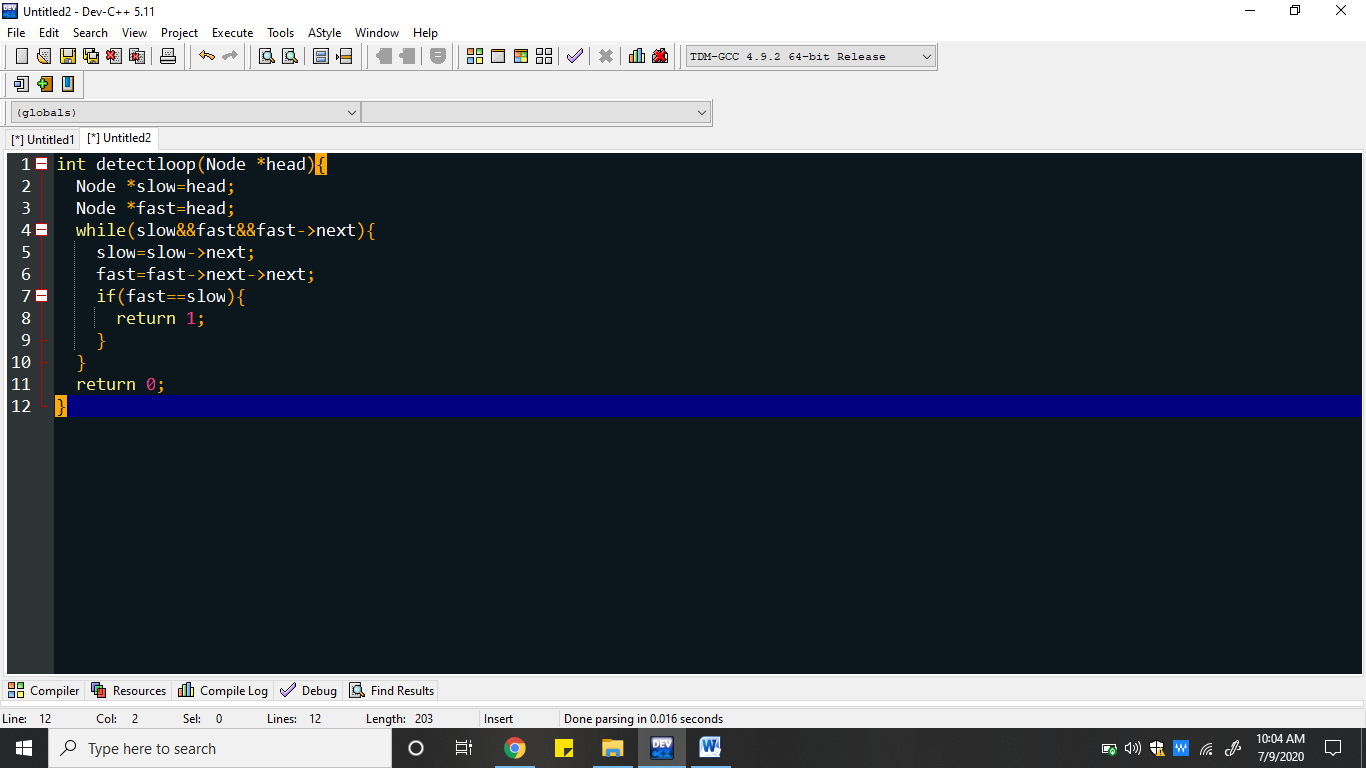
**Q1. How do you find the middle element of a singly linked list in one pass?**

**A1..** 

**Q2**. **How do you check if a given linked list contains a cycle? How do you find the starting node of the cycle?**

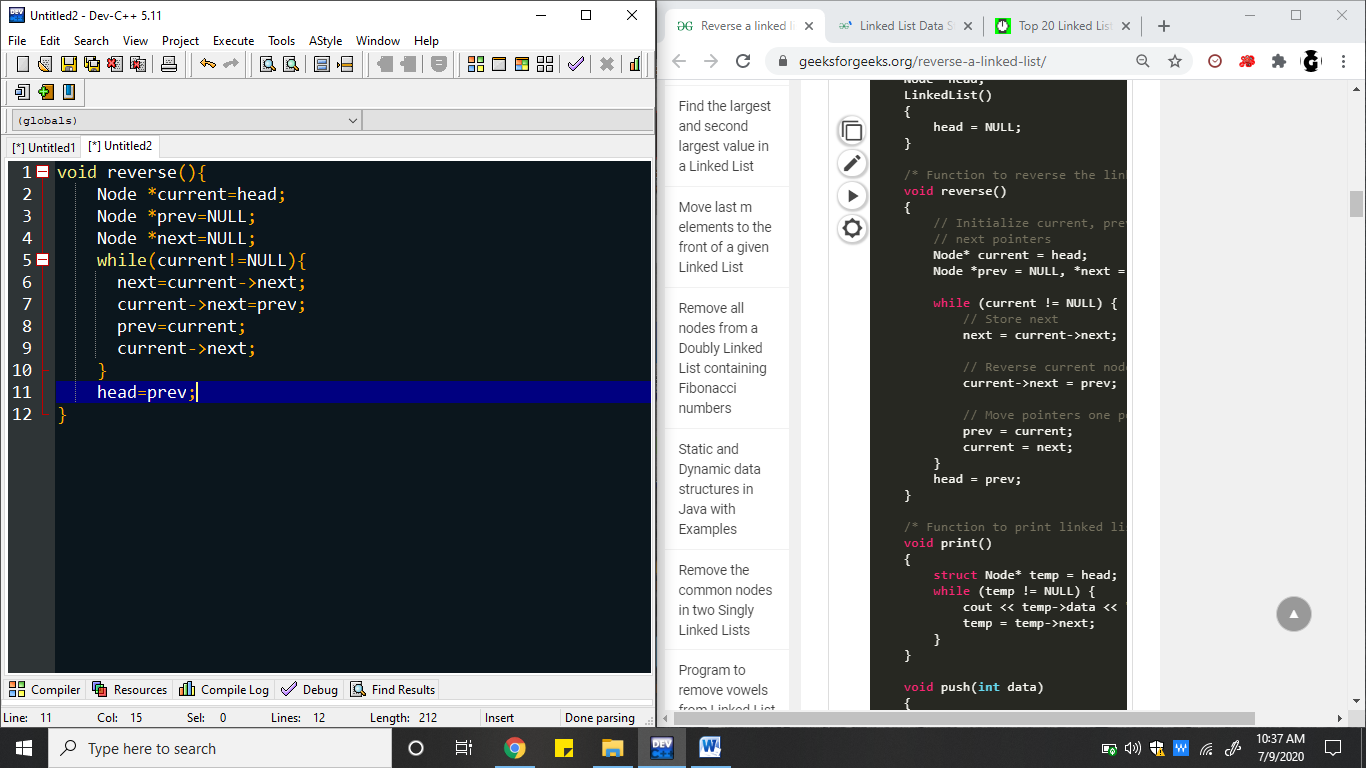
**A2.**



**The point at which fast and slow pointer meets is the first node of the loop.**

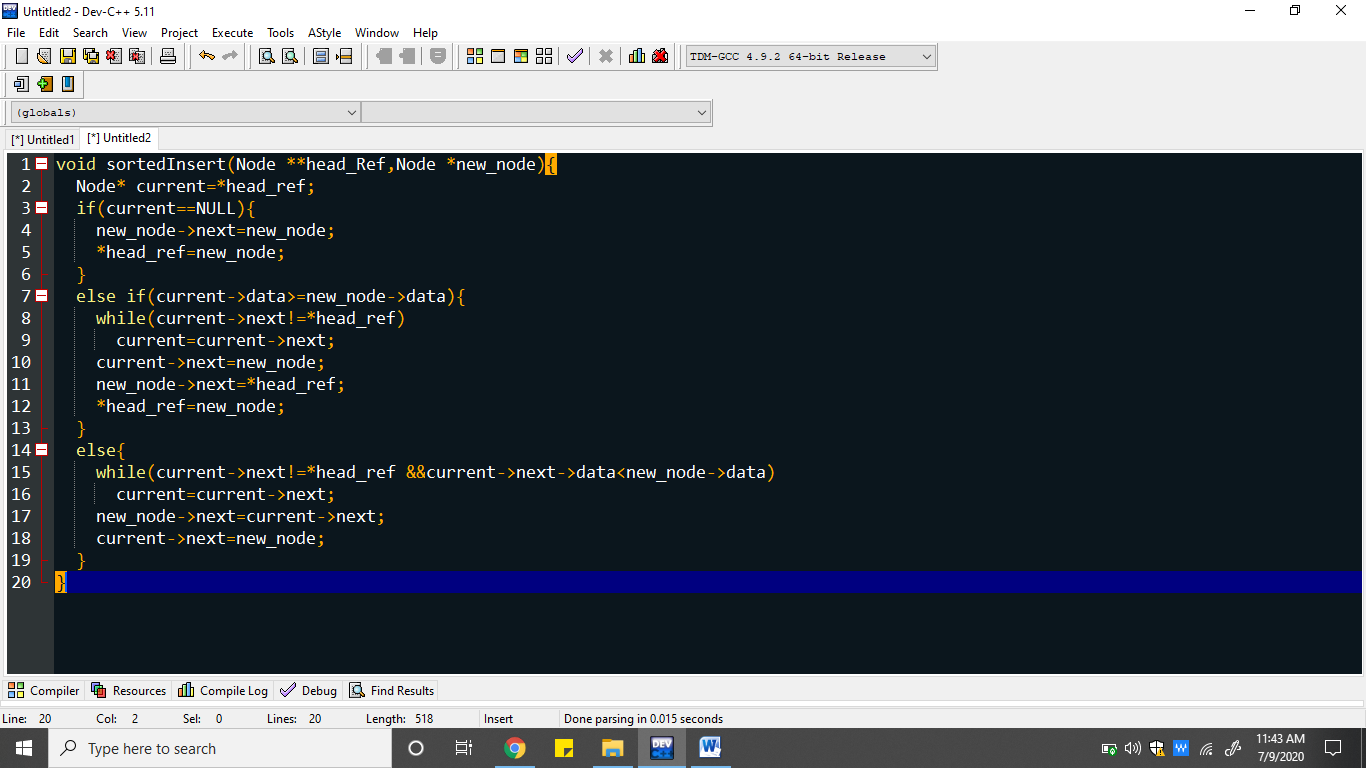
**Q3.** **How do you reverse a linked list?**

**A3.**



**Q4. Sorted insert for circular linked list**

**A4.**



**Q5. You are given two non-empty linked lists representing two non-negative integers. The digits are stored in reverse order and each of their nodes contain a single digit. Add the two numbers and return it as a linked list.**

**You may assume the two numbers do not contain any leading zero, except the number 0 itself.**

**A5. Input:** (2 -> 4 -> 3) + (5 -> 6 -> 4)

**Output:** 7 -> 0 -> 8

**Explanation:** 342 + 465 = 807. 