Northern Michigan University (Marquette Co, MI)

Time: 15 minutes.

Given the following classes:

```
class LL {                          class LLN {

 private:                            private:
  LLN *head;                          string word;
};                                    LLN *next;
                                    };
```

Write code for the following method:

```
void LL::MakeCircular ();
```

This method takes the "tail node" of a linked list (the last node in the list) and pastes it to the beginning or head of the list. At the conclusion of this method, NONE of the next pointers in the linked list should be NULL. If the list started out empty, this method should do nothing.

For example, if the list were -->A-->B-->C, after running, the list should be -->A-->B-->C-->A -->B-->C-->A-->B-->C-->...forever...

You may assume that getters and setters have already been written. Any other helper methods you use must be written by you. Use recursion only; do not use loops.

```
void LL::MakeCircular () {

 if (head) head->MakeCircular (head);
}

void LLN::MakeCircular (LLN *h) {

 if (next) next->MakeCircular (h);
 else next = h;
}
```