List Delete Largest

> This is a **pair** exercise and must be competed in your **tutorial** or **lab** with your partner.

For this activity, you will be looking at a linked list.

Download `list.h`, or copy it into your current directory on a CSE system by running

```
$ cp /web/cs1511/17s2/week11/files/list.h .
```

Make sure you understand the `list` and `node` data structures before beginning this task.

Create a file called `listDeleteLargest.c` that includes the `list.h` header file. In it, you should implement `listDeleteLargest`, a function which takes a linked list and deletes the node in the list that contains the largest value.

If there are several nodes with the same value, it should delete the first node.

i.e. if the list contains:

```
1 -> 2 -> 3 -> 4 -> 5 -> X
```

It should delete the node with value `5`, as `5` is the largest value in the list, leaving the list:

```
1 -> 2 -> 3 -> 4 -> X
```

As another example, if the list contains:

```
1 -> 2 -> 3 -> 2 -> 3 -> X
```

It should delete the first node with the value `3`, as `3` is the largest value in the list, leaving the list looking like this:

```
1 -> 2 -> 2 -> 3 -> X
`

It should have this prototype:

``` c
void listDeleteLargest (List l);
```

You should write your own tests in a separate file; `listDeleteLargest.c` should *not* contain a `main`.

To run some simple automated tests:

```
$ 1511 autotest listDeleteLargest
```

To run Styl-o-matic:

```
$ 1511 stylomatic listDeleteLargest.c
Looks good!
```

You'll get advice if you need to make changes to your code.

Submit your work with the *give* command, like so:

```
$ give cs1511 wk13_listDeleteLargest
```

Or, if you are working from home, upload the relevant file(s) to the `wk13_listDeleteLargest` activity on Give Online.