Our Project
the "method" to the madness
and some recursion
RISKS – name of the game

• What is at risk?
  – to customer?
  – to development team?
(other) RISKS

- Slipping
- Project Cancel
- Business
  - Misunder
  - Change
How do we handle the risks?

- schedule slips
- project is cancelled
- system goes bad
- defect rate so high, never used
- business misunderstood
- business changes
- false feature rich
Recursion

• mathematical definition of something that uses the item itself in the definition
  – e.g. factorial numbers
  • \( N! = N \times (N-1)! \)
Factorial Game

Who thinks up these cheesy games?

Where did he get this picture?
Factorial Game

• if my number is 1, then return 1
• if my number is not 1
  – let my neighbor do the work
  – multiply her answer by my number
Factorial Game

• Look at the bottom number shown you by left neighbor (L)
  – write it on the top of the card

• if it is one (1)
  – just show your card to your left neighbor

• if it is not one, then
  • write down your number minus one
  • show card to right neighbor (R)
  • wait
  • read bottom number on right neighbor's card
    – write down product of top number and bottom number
  • show card to left neighbor
Group work

– Now, Can you write this method? Do it!
– Show your solution to a group, check each other
Printing numbers $n$ to 1 "game"

- if my number is 1, then print 1
- if my number is not 1
  - print my number
  - then let my neighbor deal with the rest of the numbers
Printing numbers n to 1 "game"

- if my number is 1, then print 1
- if my number is not 1
  - print my number
  - then let my neighbor deal with the rest of the numbers

see the pattern?
Printing numbers $n$ to 1 "game"

- if my number is 1, then print 1
- if my number is not 1
  - print my number
  - then let my numbers

For those of you not as lazy as pipe-guy, there is another pattern to keep in mind
Printing numbers $n$ to 1 "game"

- Look at the bottom number from (L)
  - write it on card
- if it is one (1)
  - write it on board
  - just tell (L) to go ahead
- if it is not one, then
  - write it on board
  - write down $\text{your number}$ minus one
  - show card to right neighbor (R)
  - wait for (R) to tell you to go ahead
  - tell (L) to go ahead
Printing numbers \( n \) to 1 "game"

- Look at the bottom number from (L)
  - write it on card

- if it is one (1)
  - write it on board
  - just tell (L) to go ahead

- if it is not one, then
  - write it on board
  - write down \( \text{yournumber} \) minus one
  - show card to right neighbor (R)
  - wait for (R) to tell you to go ahead
  - tell (L) to go ahead
Printing numbers 1 to n "game"

- Look at the bottom number from (L)
  - write it on card

- if it is one (1)
  - write it on board
  - just tell (L) to go ahead

- if it is not one, then
  - write down yournumber minus one
  - show card to right neighbor (R)
  - wait for (R) to tell you to go ahead
  - write it on board
  - tell (L) to go ahead
find the owner of the number "game"

- if my number is the one we are looking for
  - return my name
- if my number is not the one we are looking for
  - if my neighbor is NULL, return NULL
  - otherwise, let my neighbor deal with it
find the owner of the number "game"
Recursive definitions

- LinkedList Node

```java
public class ListNode {
    private Object info;
    private ListNode next;
    ...
}
```