Recursion and pipes. Create a program which does the following

* 1. Accepts an integer, N, 1<=N<=1000
  2. Accepts a simple message from *stdin* (<100 bytes)
  3. Stores the value, N= N-1. (these 3 steps are only done once)
  4. Forks a child, writes the message to the pipe, does *wait*(NULL), then does an *exit*(0);
  5. The child examines N. If N >0, the child then decrements N and forks another child (same as parent did)
  6. When a child sees that N=0 (**before** subtracting 1), no more forking occurs. This **last** child reads from the pipe and outputs its content, then terminates.

Background info:

1. You can create a pipe 2 ways:
   1. with the linux command: **mkfifo** mypipe
   2. using the pipe() function
2. For this lab, create the pipe from inside the program, using the pipe() function
3. You can destroy the pipe with: **rm** mypipe
4. Using a pipe in commands:

**mkfifo** mypipe

**ls –l** > mypipe

**cat** < pipe