

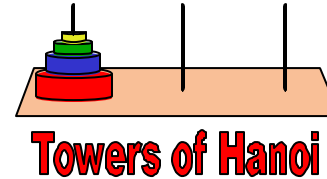
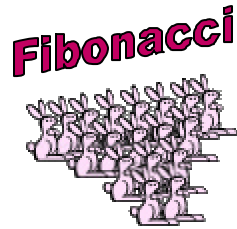
# Some New Examples for Introducing Recursion in CS1 and Beyond

Jennifer Kay

Computer Science Department, Rowan University  
kay@elvis.rowan.edu  
http://www.rowan.edu/~kay

## Old Favorites:

A brief survey of C++ textbooks shows that the classic examples are still popular



## Three Recursive Functions

• Important to emphasize that these are not intended to be the best solution to the problems, but rather that these are simple examples designed to make the topic easier to learn.

• Follow these examples with a function that returns a result (e.g. factorial, power).

### Function 1: count\_down

- Introduces the concept of a base case
- A first glimpse at recursion

```
/* Description: count_down prints the integers between num and 1, separated by spaces.
Preconditions: num>0;
Postconditions: the ints between 1 and num will be printed in order separated by spaces.
e.g. count_down (6) will print:
    6 5 4 3 2 1
*/
void count_down (int num)
{
    if (num == 1)
        cout << num;
    else
    {
        cout << num << " ";
        count_down (num-1);
    }
}
```

### Function 2: count\_up\_to\_100

- Shows that the base case is not always 0 or 1

```
/* Description: count_up_to_100 prints the integers between num and 100, separated by spaces.
Preconditions: num <= 100
Postconditions: the ints between num and 100 will be printed in order separated by spaces.
e.g. count_up_to_100(96) will print:
    96 97 98 99 100
*/
void count_up_to_100(int num)
{
    if (num == 100)
        cout << num;
    else
    {
        cout << num << " ";
        count_up_to_100 (num + 1);
    }
}
```

### Function 3: count\_down\_up

- A program most students don't believe will work
- Demonstrates how tricky recursion can be, this is so close to the first example and yet performs such a different task

```
/* Description: count_down_up prints the integers between num and 1, and back up to num separated by spaces.
Preconditions: num > 0
Postconditions: the integers between num and 1 and then back up to num will be printed in order separated by spaces. e.g. count_down_up(6) will print: 6 5 4 3 2 1 2 3 4 5 6
*/
void count_down_up (int num)
{
    if (num == 1)
        cout << num;
    else
    {
        cout << num << " ";
        count_down_up (num - 1);
        cout << " " << num;
    }
}
```