

## CSCI 104L Lecture 2: Streams and Recursion

Example usage of file streams; you'll need to include `fstream`.

```
ofstream myFile;
myFile.open("games.txt");
myFile << "1: _Minecraft" << endl;
myFile.close();
ifstream myFile2;
string line;
myFile2.open("games.txt");
getline(myFile2, line);
cout << line;
myFile2.close();
```

Example usage of string streams; you'll need to include `sstream`

```
int first;
string second;
char third;
string fourth;
stringstream ss;
ss << "1: _Minecraft";
ss >> first;
ss >> second;
ss >> third;
ss >> fourth;
cout << first << endl << second << endl << third << endl << fourth;
```

**Recursive** means "defined in terms of itself".

```
int iterativeFactorial(int n) {
    int p=1;
    for (int i=1; i <= n; i++)
        p *= i;
    return p;
}

int recursiveFactorial (int n) {
    if (n==1) return 1;
    else return n*recursiveFactorial(n-1);
}
```

What will happen if you run the following function?

```
int UCLAfactorial (int n) {
    if (n==1) return 1;
    else return UCLAfactorial(n);
}
```

What will happen if you run the following function?

```
int BruinFactorial (int n) {
    return n*BruinFactorial(n-1);
}
```

An example of (infinite) indirect recursion:



Figure 1: XKCD # 140 **Delicious**: I'm currently in the I Have Cheese phase of this cycle.

## Recursive Definitions

You can define other things recursively, not just functions.

- A string of lower-case letters is either: (1) the empty string, or (2) a letter 'a'-'z' followed by a string of lower-case letters.
- A non-negative integer is either: (1) the number 0, or (2)  $n+1$ , where  $n$  is a non-negative integer.
- A palindrome is either: (1) the empty string, or (2) a single letter 'a'-'z', or (3) a string  $xPx$ , where  $x$  is a single letter 'a'-'z', and  $P$  is a palindrome.
- A simple algebraic expression is either:
  1. A number.
  2. A variable.
  3.  $(A+B)$ , where  $A$  and  $B$  are simple algebraic expressions.
  4.  $(A*B)$ , where  $A$  and  $B$  are simple algebraic expressions.