

## 1 Objectives

- Write recursive programs

## 2 Practice With Recursion

### 2.1 Factorial

Create a new Java project in Eclipse called lab06 and import the files in

`~csci204/2009-fall/student/labs/lab06`

into lab06/src

Check your lab06 project into Subversion in the usual labs directory. Run TestFactorial. Observe the behavior of the program. Read the Factorial class and understand the logic of the program. Create a `readme.txt` file that answers the following questions.

1. If we rewrite the first branch of the `if-else` in the `factorial()` method as

```
if (n == 1)
    return 1;
```

will the final result change? Why?

2. If we rewrite the branch as

```
if (n == 0)
    return 0;
```

will the final result change? Why? Restore the program to its original state before proceeding.

3. If we revise the second branch of the `if-else` in the method as

```
else
    return 2 * n * factorial(n - 1);
```

What will happen to the final result? Can you come up with a closed form mathematical equivalent of it?

Based on the `factorial()` method given above, write a similar recursive method that adds up all the odd numbers between 1 and a given limit. Do not assume this limit is an odd number. Add a method to the Factorial class to do the work, using the signature

```
public static long addOddInts(int n);
```

Uncomment the relevant lines in `main()` to test `addOddInts()`.

## 2.2 Recursively Listing Files

Recursive algorithms are natural for solutions to problems with hierarchical structure. An example problem is listing all the files in a directory and all of its subdirectories. Since the UNIX file system is hierarchical, we should immediately think of using a recursive approach.

For this part of the lab you are to write a Java program to list all the files in a directory and, recursively, in all of its subdirectories.

Read, and run the program contained within `PrintDirectory.java`. Observe the behavior of the program.

`PrintDirectory` is a Java program that lists the names of the files and directories in a directory but does *not* recursively list the subdirectories. You are to modify the `PrintDirectory` class to recursively print the names of all the files in all subdirectories. Here are some details.

1. You must use a recursive solution.
2. File names should be printed one per line.
3. Just *before* your recursive call, print out “Entering name” where you fill in the directory name.
4. Just *after* your recursive call, print out “Leaving name” where you fill in the directory name.
5. If you print a `File` object, Java will print the *full* pathname. You can print only the file/directory name by using the `getName()` method.
6. You may check if a `File` object is a file with the method `isFile()` which returns `true` if it is a file and `false` if it is a directory. If you find it more convenient, there is a similar method `isDirectory()` that tests whether a `File` object is a directory.
7. To check your program’s output, use the following UNIX command.

```
ls -R ~/csci204/2009-fall/student/labs/lab06
```

where the `-R` option recursively lists subdirectories encountered.

Expected output of your program:

Factorial.java

Entering Hobbits

Entering Baggins

Bilbo

Frodo

Leaving Baggins

Merry

Pipin

Sam

Leaving Hobbits

PrintDirectory.java

TestFactorial.java

Entering Wizards

Gandalf

Sauraman

Leaving Wizards

Your usage of blank lines does not need to match this but the words on each line and ordering of the lines do need to match.

### 3 Upon Completion

Commit everything, including your `readme.txt`, to your subversion repository.

**Note:** Be sure to have good Javadoc comments for any methods that you have written.