Recursion – November 18, 2015

What is recursion? In simple terms, recursion is essentially breaking problems into identical smaller problems.

"*Recursion is the process of repeating in a self-similar fashion. Objects that contain self-similar smaller copies of themselves are recursive*" (http://cs.lmu.edu/~ray/notes/whatisrecursion/).

Why is recursion important? "Recursion is the way that the infinite can arise from a finite description" (http://cs.lmu.edu/~ray/notes/whatisrecursion/).

We can see recursion in everything from nature, art, computer science, math, and music – take a look around and you'll find recursion in everyday objects. Objects can be self-similar at many levels and are called fractals and can be repeated (http://cs.lmu.edu/~ray/notes/whatisrecursion/).

Some examples of recursion:

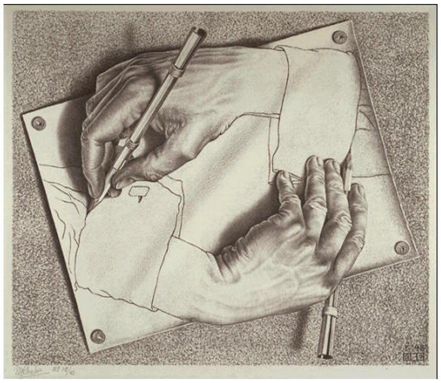
In Nature:
If you look carefully, you will find the same pattern repeat itself over and over. Look at the fern leave picture below and you can easily see that the entire leaf is composed of smaller versions (with identical structure).

A snowflake

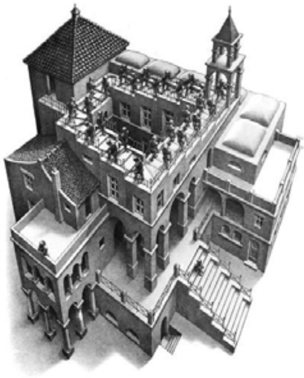
Broccoli

A Fern Leaf

In Art:
Many of the art produced by renowned artist Maurits Cornelius (M.C.) Escher has a repetitive pattern (hidden behind the drawing) that allows the artist to produce optical illusions impossible in real life. If you look at the Ascending and Descending drawing, how is it possible for the monks to continue to ascend while ending up in the same spot.



Drawing Hands, M.C. Escher



Ascending and Descending, M.C. Escher

In Performing Arts:

See if you can spot similar patterns occurring at a grandeur scale repeated by smaller groups of the performers.

Canon Rítmico (Preuniversitario Musical 2011) - watch the video on YouTube.com:
https://www.youtube.com/watch?v=TQBD4bxtZoM

In Music:

Nature Has a Good Beat, But Can You Dance To It? Listen to the NPR Story:

http://www.npr.org/2012/02/21/147180161/nature-has-a-good-beat-but-can-you-dance-to-it

In Computer Science and Math:

Recursion is commonly found in math problems and in programming. In computer science recursion is found in algorithms. Here is an example from (http://cs.lmu.edu/~ray/notes/whatisrecursion/):

```
To walk n steps:

Walk ( n steps ):

If n <= 0
    Stop
Else
    Take a single step
Walk (n-1 steps)
```

For a hands on activity that teaches recursion, use the 2014-2015 *Towers of Hanoi* Looney Challenge found in the archives.

To learn more about recursion in detail, please go to the Loyola Marymount CS Department link, http://www.cs.lmu.edu/

Let’s get started!

This activity is worth 8 points.

This Challenge will build on the *Towers of Hanoi* Challenges posted, October, 2014.

Beginners

1) After learning about recursion. Ask your students to look for examples of recursion in the environment around them. Examples can be found in nature, art, programming games, etc. What examples did they find?

2) Ask your students to draw, paint, or use a computer to show an example of recursion (for younger students you can ask them to create snowflakes out of paper, draw a flower, etc.)

Intermediate

1) Follow the instructions above for beginners, and complete the same tasks.

2) Ask your students to describe recursion used in a programming game they are familiar with (e.g. Lightbot or Foos)

Experienced

1) Follow the instructions above for beginners and intermediate students and complete the same tasks

2) In addition, ask your students to go online and research recursion. What are two examples of recursion used in math and/or computer science? Write a short computer program that uses recursion (similar to the example described above).

3) For extra credit: Students can research M.C. Escher and how he used recursion in his drawings and why he was fascinated with mathematics. What is their favorite M.C. Escher drawing?

How to earn points:

1) If you haven't registered your class, please go to cs.montana.edu/looney-challenge and click on the "Register for Looney Challenges" link.

2) Discuss with your class the difficulty of the activity. What did they learn? How difficult was the activity? Do they understand the concept and how it relates to computational thinking?

3) Describe in your email what happened during the activity? Did your students understand the concept(s)? Email your description to looneychallenges@gmail.com

4) If you want to attach an example, photographs of students working, or video of student's outcomes, please send them as an attachment.

5) We will send you a confirmation and provide you your point total for the activity and your total points for Looney Challenges.

All Looney Challenges can be completed at any time during the 2015-2016 school year. All Challenges are due, June 30, 2015.

Questions? Please send an email to looneychallenges@gmail.com or call Sharlyn Izurieta at (406) 994-4794.

Thank you for participating in Looney Challenges!

If you still need to register your class, please go to the registration page by clicking HERE.

DEADLINE is June 30th, 2016.