

# Advanced Programming (CSE 399)

## Homework Assignment 2

Due Monday, January 24, at noon

1. Choose *one* of the following two tasks, depending on whether you are having success getting the graphics library to work with hugs:

- (a) The *towers of Hanoi* is a simple recursive game. (You've probably seen it before, so I won't provide a lot of detail here; if it is not familiar, you may want to read a little about it on the web to get some intuition.) The purpose is to move a stack of  $n$  discs from one peg to another, using a third peg as "temporary storage" and respecting the constraint that no disc is ever placed on top of a smaller one. It is played as follows:

To move  $n$  discs (stacked in increasing size) from peg  $a$  to peg  $b$  using peg  $c$  as temporary storage...

- i. move  $n$  discs from  $a$  to  $c$  using  $b$  as temporary storage
- ii. move the top disc from  $a$  to  $b$
- iii. move  $n$  discs from  $c$  to  $b$  using  $a$  as temporary storage.

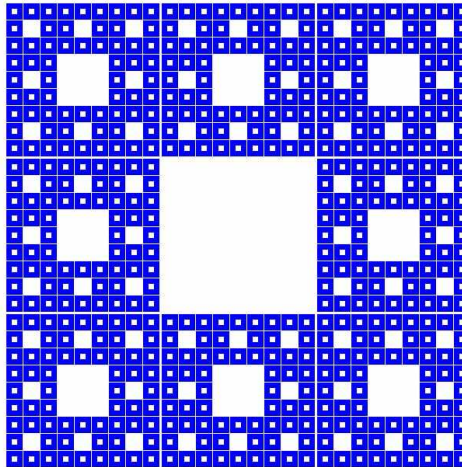
Write a function `hanoi :: Int -> String -> String -> String -> IO()` that plays towers of Hanoi. For example, when invoked like this

```
hanoi 2 "a" "b" "c"
```

it should yield an action that, when executed, prints this:

```
move disc from a to c
move disc from a to b
move disc from c to b
```

- (b) The *Sierpinski Carpet* is a recursive figure with a structure similar to the Sierpinski Triangle discussed in Chapter 2:



Write a function that displays this figure on the screen.

(Technical hint: If you encounter problems with hugs running out of memory while drawing your carpet, you may need to add a command-line switch like `-h1000k` to increase the heap size.)

2. Work problems 5.1, 5.3, 5.5, 7.1, and 7.2 from SOE.
3. Extra credit: Work problem 7.5 from SOE.

**Submission instructions:**

- Collect your solutions into a *single file* of Haskell source code, with clearly marked sections for the solutions to the various exercises.

Name this file by concatenating *your last name* with the extension “`.hs`” (note that this is different from the instructions for hw1 — this way makes it much easier to unpack all the emails).

- For each exercise, write a small “test harness” that demonstrates that it works (by drawing a sierpinski carpet on the screen or dumping the towers of hanoi to the standard output in the case of the first problem; by printing the results of a few tests on the standard output for the rest). Define a single `main` action that performs *all* of the tests.
- Make sure this file is accepted by Hugs without errors and follows the other rules in the Style Guide on the course web page.  
In particular, please put your name in a comment at the top of the file.
- Email this file (preferably as an attachment) to `bcpierce@cis.upenn.edu`.