

Harvard University
Computer Science 20
In-Class Problems 19
Friday, March 11, 2016

Author: Crystal Chang, Tom Silver

Review

PROBLEM 1

Let $\{A, B, C, D, E, F, G\}$ be an alphabet of musical notes. A **song** is defined by some sequence of notes in the alphabet. (So for example, assume only quarter notes on one instrument at one volume is allowed.) For each of the following, state whether the set is finite, countably infinite, or uncountable. Justify your answer by describing a bijection to a set of the same cardinality with known countability.

1. The set of all songs with exactly 100 notes.
2. The set of all songs with at least 100 notes.
3. The set of all finite length songs.
4. The set of all (possibly infinite length) songs.
5. The set of all (possibly infinite length) songs consisting of only B notes.

Solution.

TODO

PROBLEM 2

In the “Internet Directed Graph” (IDG), vertices represent webpages, and an arrow from webpage u to webpage v indicates that webpage u contains a hyperlink to webpage v .

(A) Is the relation corresponding to the IDG reflexive? Symmetric? Transitive?

(B) What property of a vertex u in the IDG could we use as an estimate for webpage u ’s popularity? Explain your answer in two sentences or less.

(C) Let $Access(u) = \{v \mid \text{there is a path from } u \text{ to } v \text{ in IDG}\}$. Is the following statement true or false?

$$Access(u_1) = Access(u_2) \implies u_1 = u_2.$$

If it is true, provide a brief proof. If it is false, provide a counterexample.

Solution.

- (A) It is none of these.
- (B) The vertex's in-degree corresponds to the number of webpages that link to the vertex. Other well-justified answers may also be accepted, e.g. average distance from other vertices in the connected component.
- (C) False. For example, consider two distinct webpages that have no hyperlinks. Then both have $Access(\cdot) = \emptyset$, but the vertices are distinct.

PROBLEM 3

(Bonus) The set of Pythagoras trees:

The shape of Pythagoras tree is built by recursively feeding the axiom through the production rules. Each character of the input string is checked against the rule list to determine which character or string to replace it with in the output string. In this example, a '1' in the input string becomes '11' in the output string, while '[' remains the same.

- variables: 0,1
- constants: [,]
- axiom: 0
- rules: $(1 \rightarrow 11), (0 \rightarrow 1[0]0)$

(A) Give a recursive definition of the Pythagoras tree.

(B) List the first 4 elements.

(C) This string can be drawn as an image by using turtle graphics, where each symbol is assigned a graphical operation for the turtle to perform. For example, in the sample above, the turtle may be given the following instructions:

- 0: draw a line segment ending in a leaf
- 1: draw a line segment
- [: push position and angle, turn left 45 degrees
-]: pop position and angle, turn right 45 degrees

Applying the graphical rules listed above to the earlier recursion, we could get the graphical representation of axiom, 1st and 2nd recursion as the following :



Draw out the graphical representation of the 3rd and 4th recursion.

Solution.

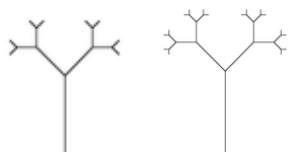
(A)

- Base Case: $0 \in E$

- Constructor Rule: $(1 \rightarrow 11), (0 \rightarrow 1[0]0)$
- Nothing else (generally implicit): Nothing is in the set of Pythagoras tree unless it is obtained from the base case and constructor rule.

(B)

- axiom: 0
- 1st recursion: $1[0]0$
- 2nd recursion: $11[1[0]0]1[0]0$
- 3rd recursion: $1111[11[1[0]0]1[0]0]11[1[0]0]1[0]0$



(C)