11-6

CS 53, Fall 2017

Due Nov. 8 at 2:59 pm

- Do problem 8.6.1 in the textbook.

- Also do the following problem. The problem has its own stencil. It consists of three subproblems. The first subproblem, in which you write a procedure, is auto-graded. The other two problems require some coding but also some mathematical reasoning. I found it most convenient to enter the code directly into the Python REPL to get my answers, but you should include your code in the stencil and use the usual `cs053_submit` script to hand it in. That is, once you have included the code you used in solving all three problems, be sure to run the script so your entire stencil is handed in. Also, please print it out and include with it your written explanations.

Solvability of *Lights Out*

Problem 1: 1. Write a procedure `button_vectors(n)` that, given a positive integer n, returns n^2 button vectors over $GF(2)$, one for each pair (i, j) in $\{0, 1, 2 \ldots, n-1\} \times \{0, 1, 2 \ldots, n-1\}$.

Here is an illustration of how the procedure is used:

```
>>> button_vectors(2)
{(0, 1):Vec({(0, 1), ..., (1, 1)},{(0, 1):one, (0, 0):one, (1, 1):one}),
 (1, 0):Vec({(0, 1), ... (1, 1)},{(1, 0):one, (0, 0):one, (1, 1):one}),
 (0, 0):Vec({(0, 1), ..., (1, 1)},{(0, 1):one, (1, 0):one, (0, 0):one}),
 (1, 1):Vec({(0, 1), ..., (1, 1)},{(0, 1):one, (1, 0):one, (1, 1):one})}
```

This procedure is used in the following two subproblems.

2. Use procedures written or provided in this class to find the set of buttons to push (as a set of pairs (i, j)) to solve the following, or report that no such solution exists.

   (a) For $n = 3$, starting with the configuration {`(1,1):one, (1,2):one`}.

   (b) For $n = 5$, starting with the configuration {`(2,2):one`}.

   (c) For $n = 9$, starting with the configuration {`(3,4):one, (6,7):one`}.

   Print out and hand in your code, and explain your answers.

3. For $n \times n$ *Lights Out*, there are 2^{n^2} possible initial configurations. You are to determine, for each of the following values of n, how many of these initial configurations are *not* solvable, i.e. there is no sequence of buttons to press that would lead to all lights being off (all entries being zero).

   You should use procedures written or provided in this class to calculate the answers. Print out and hand in your Python code, and use words to explain and justify your answer.

(a) $n = 2$

(b) $n = 3$

(c) $n = 4$

(d) $n = 5$

(e) $n = 20$

Justify your answer.