Lecture 2

Wednesday, January 11, 2023 4:32 PM

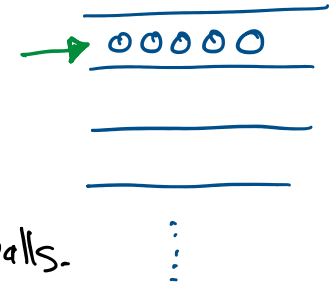
Studying various models of computation

* We have a function
  $f: A \to B$

* We want to "compute" f.
* The computation involves a "finite" set of operations (chosen from a finite and predefined set of operations) that are "applied" to x, generating output f(x).
  * The set of predefined operations and the way we give the input and take the output define our computational model.
  * We say f is computable in our model if we can find a finite set of operations (a program) that gives the right output for $\forall x \in A$.

The SCRAMBLE machine

* the machine has an infinite number of rows. Some of these rows may contain a finite number of balls.



* There is a lever that points at a row.
* We have some operations to manipulate

the balls or the lever.

* The machine has a single bit of memory, which we denote MEM (MEM = false or true)

Operations

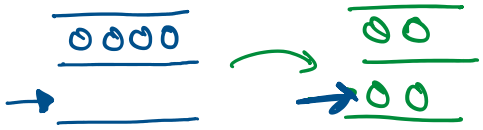
LOWER_LEVER: the lever goes to one row below it

RAISE_LEVER: " " " " " " above it.

CHECK_EMPTY: checks whether the row that the lever points to is empty. If so, then sets MEM=TRUE. Otherwis sets MEM = FALSE.

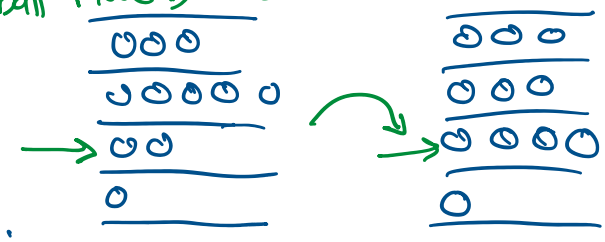
RESET_BALLS: puts all the balls in the first row.

SCRAMBLE_DOWN:



the balls in rows above the lever will start to "fall down" until #balls in each row (above the lever) is smaller or eqcal to its bottom row.

* At the end, if any ball moved, MEM = TRUE, o.w. MEM = FALSE

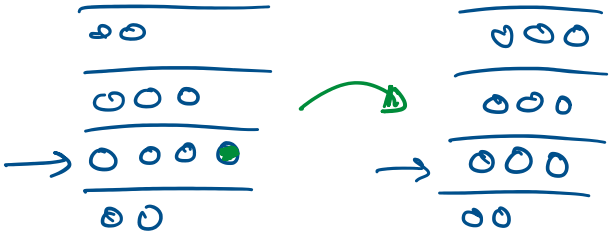


SCRAMBLE_UP:

the balls in rows on or above the lever will start to "rise" until #balls in each row on or above the lever is smaller or eqcal

the lever is smaller or equal to its above row

* MEM is set like above.



RETURN_FALSE_IF_MEM_FALSE :

" " " " TRUE :

" TRUE " " " "

" " " " FALSE:

these terminate the program only if the condition holds.

LOOP: LOOP : loops over the ops forever.

Example: Assume $X \in \mathbb{N}$ and we want to compute whether X is even or not.

$$f(X) = \begin{cases} \text{TRUE}, & X \text{ is even} \\ \text{FALSE}, & X \text{ is odd} \end{cases}$$

Assume we put the same number of balls as X on the first row, and the lever also points to first row. write a program that computes f?

Write a program that computes f?

LOWER_LEVER
SCRAMBLE_DOWN
SCRAMBLE_UP
RETURN_TRUE_IF_MEM_FALSE
RETURN_FALSE_IF_MEM_TRUE

Write a program that determines if $x \in \mathbb{N}$ is a prime number.

$$f(x) = \begin{cases} \text{TRUE} & x \text{ is prime} \\ \text{FALSE} & \text{o.w.} \end{cases}$$

Is there a simple f that cannot be implemented with our scramble machine.
