Solution to Problem 12

Problem. In class, we described a Turing machine that computes $g(n) = n+1$. In Homework 9, you designed a Turing machine that computes a function $f(n)$ which is equal to $n - 3$ when $n > 3$ and to 0 otherwise.

In class, we described the general algorithm for designing a Turing machine that computes the composition of two functions. The assignment is to use this general algorithm to design a Turing machine that computes the composition $g(f(n))$. Trace, step-by-step, on an example, how your Turing machine works. For example, you can take as input $n = 1$.

Reminder: The Turing machine for computing $g(n) = n + 1$ for a unary input n is based on the following idea:

- we go step-by-step until we find the first blank space,
- then, we replace this blank space with 1 and go back.

This machine has the following rules:

- start, $- \to$ R, working
  (we start going to the right)
- working, $1 \to$ R
  (we see 1, so we continue going),
- working, $- \to$ 1, L, back
  (we see a blank space, so we replace it with 1 and start going back)
- back, $\to$ L
  (while we see 1s, we continue going back)
- back, $- \to$ halt
  (once we reach the very first cell, we stop).

The Turing machine for computing $f(n)$ is based on the following idea:

- we go step-by-step until we find the first blank space,
- then, we go back, replace the last three 1s with blanks, and go back all the way.

We need to take special care of the case when $n < 3$.

Solution. The resulting Turing machine takes the following form:

- start, – → R, moving_1
- moving_1, – → L, start_2
- moving_1, 1 → R
- moving_1, – → 1, L, back_1
- back_1, 1 → L
- back_1, – → start_2
- start_2, – → R, working_2
- working_2, 1 → R
- working_2, – → L, delete1st_2
- delete1st_2, 1 → –, L, delete2nd_2
- delete1st_2, – → halt
- delete2nd_2, 1 → –, L, delete3rd_2
- deket2nd_2, – → halt
- delete3rd_2, 1 → –, L, back_2
- delete3rd_2, – → halt
- back_2, 1 → L
- back_2, – → halt

Let us trace it for $n = 1$;

$\underline{-}$	1	–	–	–	–	–	...	start
–	$\underline{1}$	–	–	–	–	–	...	moving_1
–	1	$\underline{-}$	–	–	–	–	...	moving_1
–	$\underline{1}$	1	–	–	–	–	...	back_1
$\underline{-}$	1	1	–	–	–	–	...	back_1
$\underline{-}$	1	1	–	–	–	–	...	start_2
–	$\underline{1}$	1	–	–	–	–	...	working_2
–	1	$\underline{1}$	–	–	–	–	...	working_2
–	1	1	$\underline{-}$	–	–	–	...	working_2

−	1	$\underline{1}$	−	−	−	−	. . .	delete1st$_2$
−	$\underline{1}$	−	−	−	−	−	. . .	delete2nd$_2$
$\underline{-}$	−	−	−	−	−	−	. . .	delete3rd$_2$
$\underline{-}$	−	−	−	−	−	−	. . .	halt