

Pre-Interview Solutions

Exercise 1. We define the *integer square root* of a positive integer n as a positive integer r such that

$$r^2 \leq n < (r+1)^2$$

```
1      def iroot(n):
2          """
3          PRECONDITION: n is an integer >= 0
4          POSTCONDITION: return integer square root of n.
5          """
6          r = 0
7          while (r+1)**2 <= n:
8              r = r + 1
9          return r
```

Prove the correctness of the above algorithm.

Proof. Let r_i represent the value of the variable `r` at the start of the i th iteration of the while loop on line 7. For some positive integer i , define the loop invariant

$$P(i) : r_i \text{ is a positive integer and } r_i^2 \leq n.$$

Now, we proceed with simple induction to show that for positive integers i , at the beginning of the i th loop iteration, then $P(i)$ holds true. At the start of the first iteration, namely $i = 1$, $r_i = 0$ and $r_i^2 \leq n$ so $P(1)$ holds.

Next, suppose that for some i , $P(i)$ is true at the start of the i th iteration. It will be shown that $P(i+1)$ is true at the start of the $i+1$ th iteration. If $(r_i+1)^2 > n$, the loop terminates and there is nothing to prove, so assume that $(r_i+1)^2 \leq n$. `r` is incremented by 1, so $r_{i+1} = r_i + 1$ and the program arrives at the beginning of the $i+1$ th iteration. Clearly r_{i+1} is a positive integer and $r_{i+1}^2 \leq n$, so $P(i+1)$ is true. By induction, we can conclude that $P(i)$ holds at the start of the i th iteration, for any positive integer i .

To show that the loop terminates, define the loop variant $v = n - (r+1)^2$. v is an integer since n and r are integers, and decreases each loop iteration since r increases. Thus there exists an iteration j where $v < 0$, that is, $(r+1)^2 > n$, so the loop condition is not met and the loop is terminated.

By the loop invariant, r_j is a positive integer such that $r_j^2 \leq n$ but also $n < (r_j+1)^2$, so r_j is the desired integer square root of n , which gets returned in line 9. Therefore the algorithm works as intended and the proof is done.

□

Exercise 2. Let Σ be an alphabet. The *reversal*, denoted by $\text{rev}(w)$ of a string $w = w_1w_2 \cdots w_l$ is defined by

$$\text{rev}(w_1w_2 \cdots w_l) = w_l \cdots w_2w_1$$

where $w_i \in \Sigma$. For a language L over Σ , the *reversal* of the language L is defined by

$$L^\leftarrow = \{\text{rev}(w) : w \in L\}$$

Show using structural induction that if L is regular then so is L^\leftarrow .

Proof. We proceed with structural induction over the set of regular languages. If $L = \emptyset$, then $L^\leftarrow = \emptyset$ is regular. If $L = \{a\}$, for some $a \in \Sigma$, then $L^\leftarrow = \{a\}$ is regular.

Let L, M be regular languages over Σ such that their reversals are also regular. We will prove the following claims:

(a) $(L \cup M)^\leftarrow = L^\leftarrow \cup M^\leftarrow$

Let $w \in (L \cup M)^\leftarrow$. Then $w = \text{rev}(v)$, where $v \in L \cup M$. It follows that $v \in L$ or $v \in M$ and $w = \text{rev}(v) \in L^\leftarrow$ or $w \in M^\leftarrow$ so $w \in L^\leftarrow \cup M^\leftarrow$. Thus we have $(L \cup M)^\leftarrow \subseteq L^\leftarrow \cup M^\leftarrow$. For the reverse inclusion, simply reverse each step above.

(b) $(L + M)^\leftarrow = M^\leftarrow + L^\leftarrow$

Let $w \in (L + M)^\leftarrow$. Then $w = \text{rev}(xy)$, where $x \in L$ and $y \in M$. Write $x = x_1 \cdots x_l$ and $y = y_1 \cdots y_m$. Notice that $w = y_m \cdots x_1x_l \cdots x_1 = \text{rev}(y)\text{rev}(x)$. Since $\text{rev}(x) \in L^\leftarrow$ and $\text{rev}(y) \in M^\leftarrow$, we have $w \in M^\leftarrow + L^\leftarrow$. To obtain the reverse inclusion, reverse each step.

(c) $(L^*)^\leftarrow = (L^\leftarrow)^*$

If $w \in (L^*)^\leftarrow$, then $w = \text{rev}(v)$ for some $v \in L^*$. We write $v = v_1 \cdots v_l$, where $v_i \in L$, and apply the argument from (b) to get that $w = \text{rev}(v_l) \cdots \text{rev}(v_1)$, so $w \in (L^\leftarrow)^*$ and therefore $(L^*)^\leftarrow \subseteq (L^\leftarrow)^*$. Once again, reverse the steps above to conclude that $(L^*)^\leftarrow = (L^\leftarrow)^*$.

Finally, since L^\leftarrow and M^\leftarrow are regular, it follows that $(L \cup M)^\leftarrow, (L + M)^\leftarrow, (L^*)^\leftarrow$ are regular. By the principle of structural induction, the reversal of regular languages are also regular. \square