

Harvard University  
Computer Science 20  
In-Class Problems 16  
Friday, March 04, 2016

Author: Ben Zheng, Crystal Chang

## Executive Summary

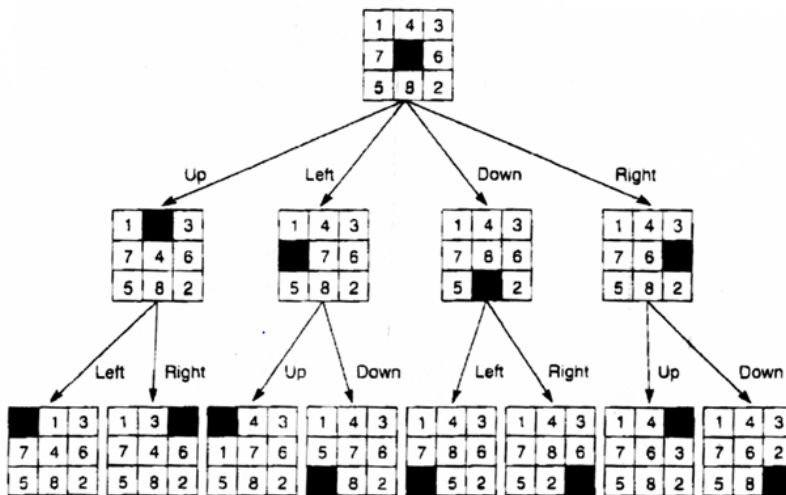
### PROBLEM 1

We have a puzzle look like the following graph. What's the size of the state space after 2 moves?

|   |   |   |
|---|---|---|
| 1 | 4 | 3 |
| 7 |   | 6 |
| 5 | 8 | 2 |

#### Solution.

The size of the state space is 8.



### PROBLEM 2

Use loop invariant to prove correctness property that  $y=c$  ( $c > 0$ ) after the following loop terminates.

- $x=c$ ;  $y=0$ ;
- while ( $x > 0$ ):  
 $\quad x --$   
 $\quad y ++$

- (A) Construct a loop invariant for the proof.
- (B) Use induction to prove that the loop preserves the invariant.
- (C) Use loop invariant to prove correctness property that  $y=c$  ( $c > 0$ ) after loop terminates.

**Solution.**

(A)

- Let state set  $S=N * N * N$ : the values of  $(x, y, c)$
- $(x, y, c) \rightarrow (x - 1, y + 1, c)$  in each iteration of loop.
- Let  $P(x, y, n) \equiv "x+y=c"$
- $y++$

(B)

- Base case: loop invariant  $x+y=c+0=c \rightarrow P(c,0,c)$  holds.
- Induction step:  
 Assume loop invariant holds after  $k$  iterations:  
 $x=c-k; y=k;$   
 after the  $(k+1)$  th iteration,  $y=k+1, x=c-k-1$   
 And  $x+y=k+1+c-k-1=c$   
 Therefore, the loop preserves the invariant  $P(x,y,n)$ .

(C) After final iteration:  $x=0;$

we also know our loop invariant holds:  $x+y=c$ . Therefore,  $y=c$ .

### PROBLEM 3

Consider the following piece of code ( $n > 0$ ):

- $y=0; i=0$
- while ( $i < n$ ):  
 $y+ = 2^i$   
 $i++$

- (A) Compute the value of  $y$  after 0th,1th,2nd,3rd iteration and guess what would be the value of  $y$  after the loop termination?
- (B) Take use of what you get in (A) to construct a loop invariant.
- (C) Use induction to prove that the loop preserves the invariant.
- (D) Use loop invariant to prove correctness property that  $y=c$  ( $c > 0$ ) after loop terminates.

**Solution.**

(A)

- iteration 0:  $y_0 = 0 = 2^0 - 1$
- iteration 1:  $y_1 = 2^0 = 1 = 2^1 - 1$
- iteration 2:  $y_2 = 2^0 + 2^1 = 1 + 2 = 3 = 2^2 - 1$
- iteration 3:  $y_3 = 2^0 + 2^1 + 2^2 = 1 + 2 + 4 = 7 = 2^3 - 1$
- iteration n:  $y_n = 2^0 + 2^1 + 2^2 + \dots + 2^{n-1} = 2^n - 1$

(B)

- Let state set  $S = N * N * N$ : the values of  $(y, i, n)$
- $(y, i, n) \rightarrow (y + 2^i, i + 1, n)$  in each iteration of loop.
- Let  $P(y, i, n) \equiv y = 2^i - 1$

(C)

- Base case:  $i=0$ :  $y_0 = 0 = 2^0 - 1 \rightarrow P(0,0,n)$  holds.
- Induction step:  
Assume that at the start of the k-th iteration  $y_k = 2^k - 1$   
Then, at the start of the (k+1)-th iteration we will have:  
 $y_{k+1} = y_k + 2^k = 2^k - 1 + 2^k = 2 * 2^k - 1 = 2^{k+1} - 1$  Q.E.D.

(D) When the loop terminates  $i=n$ . Thus after the loop execution we have:  $y = 2^n - 1$