CS21201 Discrete Structures
Tutorial Solutions
Recursive Constructions, Loop Invariance

---

1. What does the following function return on input n? Also argue that the function terminates for n > 1.
```
int h ( int n ) {
      if (n <= 0) return -1; /* Error condition */
      if (n % 2 == 1) return 0; /* n is odd */
      return 1 + h(n*(n+1)/2); /* n is even */
}
```
Solution: If n<=0, function returns -1.
Let $n = 2^k l$, where k is a non-negative integer and l is an odd positive integer.
If k=0, function returns 0.
If k>0, n+1 is odd, n*(n+1)/2 can be written as $2^{k-1}m$, where m is odd, and the function recursively adds 1. Therefore, the function returns k.
Hence, the function terminates for n > 1.

2. What does the following function return upon the input of two positive integers a,b? Prove it.
```
int f(int a, int b) {
  int x, y, u, v;
  x = u = a;
  y = v = b;
  while (x != y) {
    if (x > y) {
      x = x - y;
      u = u + v;
    } else {
      y = y - x;
      v = u + v;
    }
  }
  return (u + v) / 2;
}
```
Solution: The function returns $lcm(a, b)$. The loop maintains the two invariances
$$gcd(x, y) = gcd(a, b)$$
$$vx + uy = 2ab$$
Show that this is true at all the times when the loop condition is checked. The loop terminates when $x = y$. At that time, $gcd(a, b) = gcd(x, y) = x = y$,and so $(v + u)gcd(a, b) = 2ab$. But then $(u + v)/2 = ab/gcd(a, b) = lcm(a, b)$.

3. Let S be the subset of the set of ordered pairs of integers defined recursively by
Basis step: (1, 2) ∈ S.
Recursive step: If (a, b) ∈ S, then (a + b, b) ∈ S, (a - b, b) ∈ S and (b, a) ∈ S.
    a. Does S contain (20, 23)?
    b. Does S contain (357, 819)?
    c. Find an invariance between any two ordered pairs (p, q) and (x, y), where both belong to S.

Solution: $gcd(a, b)$ is the invariant as

$$gcd(a, b) = gcd(a - b, b) = gcd(a + b, b) = gcd(b, a).$$

a. S contains (20,23) as $gcd(20, 23) = gcd(1, 2) = 1$
b. S does not contain (357, 819) as $gcd(357, 819) = 21 \neq gcd(1, 2)$.
c. $gcd(a, b)$ as already shown.

4. a) Start with a point S(a,b) of the plane with 0 < a < b, we generate a sequence (x_n, y_n) of the points according to the rule

$$x_0 = a,\ y_0 = b, x_{n+1} = \sqrt{x_n y_{n+1}},\ y_{n+1} = \sqrt{x_n y_n}$$

Prove that there is a limiting point with x = y. Find this limit.
b) Take above definition of S(a,b) with:

$$x_0 = a,\ y_0 = b, x_{n+1} = \frac{2x_n y_n}{x_n + y_n},\ y_{n+1} = \frac{2x_{n+1} y_n}{x_{n+1} + y_n}$$

Prove that there is a limiting point with x = y. Find this limit.
Solution:

a) Consider the value $x_k^2 y_k$, which does not change for any $k \in N$. This can be shown as follows:

$$x_{k+1}^2 y_{k+1} = x_k y_{k+1} y_{k+1} = x_k y_{k+1}^2 = x_k x_k y_k = x_k^2 y_k = a^2 b$$

Now, y_{n+1} is the G.M. of x_n and y_n and hence, lies between them. Similarly, x_{n+1} is the G.M. of x_n and y_{n+1} and lies between them. Therefore, we can observe that the gap between x_{n+1} and y_{n+1} decreases in each iteration.

Let the limit be $z = x_\infty = y_\infty$. Then, $x_\infty^2 y_\infty = z^3 = x_0^2 y_0 = a^2 b$.

Therefore, the limit $z = \sqrt[3]{a^2 b}$.

b) Consider the value $\frac{1}{x_k} + \frac{2}{y_k}$, which does not change for any $k \in N$. This can be shown as follows:

$$\frac{1}{x_{k+1}} + \frac{2}{y_{k+1}} = \frac{1}{x_{k+1}} + \frac{1}{x_{k+1}} + \frac{1}{y_k} = \frac{2}{x_{k+1}} + \frac{1}{y_k} = \frac{1}{x_k} + \frac{2}{y_k} = \frac{1}{a} + \frac{2}{b}$$

y_{n+1} and x_{n+1} are harmonic means and the rest can be argued as in a).

Let the limit be $z = x_\infty = y_\infty$. Then, $\frac{1}{x_\infty} + \frac{2}{y_\infty} = \frac{3}{z} = \frac{1}{x_0} + \frac{2}{y_0} = \frac{1}{a} + \frac{2}{b}$.

Therefore, the limit $z = \frac{3ab}{b+2a}$.

5. You have six integers $a1, a2, a3, a4, a5, a6$ arranged in the clock-wise fashion on a circle. Their initial values are $1, 0, 1, 0, 0, 0$ respectively. You then run a loop, each iteration of which takes two consecutive integers (that is, $(a1,\ a2)\ or\ (a2,\ a3)\ or \cdots or\ (a6, a1)$), and increments both the chosen integers by 1. Your goal is to make all the six integers equal. Propose a way to achieve this using

the above loop (that is, specify which pairs you choose in different iterations), or prove that this cannot be done.

Solution: Look at the alternating sum $a1 - a2 + a3 - a4 + a5 - a6$. This is initially equal to 2. The given operation does not change this sum (invariance), so it can never attain the value 0.

Practice Problems Solutions
Recursive Constructions, Loop Invariance

1. Let $s(n, m)$ denote the number of permutations of $1, 2, 3,..., n$ that have exactly m cycles. For example, the permutation $3, 1, 6, 8, 2, 5, 7, 4$ $(for\ n = 8)$ has three cycles $(1, 3, 6, 5, 2), (4, 8), (7)$. The numbers $s(n, m)$ are called Stirling numbers of the first kind. Prove that $s(m, n) = s(m - 1, n - 1) + (m - 1)\, s(m - 1, n)$.
   Solution: We consider forming a permutation of m numbers from a permutation of $m - 1$ numbers by adding an additional number:

   Case 1: Forming a single cycle - We can add the new number as a cycle by itself (singleton cycle). This increases the number of cycles by 1. The number of such permutations is $s(m - 1, n - 1)$.

   Case 2: Inserting the new number into an existing cycle - Alternatively, we can insert the new number into any of the existing cycles in a permutation of $m - 1$ numbers. There are $m - 1$ ways to do this, and since the number of cycles remains the same, this accounts for $(m - 1)s(m - 1, n)$.

2. Prove using the theory of loop invariance that the following function prints d, u, v, where d = gcd(x, y) = ux + vy with d ∈ N and u, v ∈ Z. Assume that both the arguments x and y are supplied with positive values.

```
void egcd ( unsigned int x, unsigned int y ) {
      int a1, a2, u1, u2;
      a1 = x; a2 = y; u1 = 1; u2 = 0;
      while (a1 != a2) {
            if (a1 > a2) { a1 = a1 - a2; u1 = u1 - u2; }
            else { a2 = a2 - a1; u2 = u2 - u1; }
      }
      printf("%d, %d, %d\n", a1, u1, (a1 - u1 * x) / y);
}
```

   Solution: The loop of the given function maintains two invariance properties.
   (1) $gcd(a_1, a_2) = gcd(x, y)$.
   (2) There exist integers v1, v2 such that $a_1 = u_1x + v_1y$ and $a_2 = u_2x + v_2y$.

Initially, $a_1 = x$ and $a_2 = y$, so Invariance (1) is true. It is maintained because $gcd(a_1, a_2) = gcd(a_1 - a_2, a_2) = gcd(a_1, a_2 - a_1)$.

Given $a_1, a_2, u_1, u_2 \in Z$, we can always find unique values for $v_1 = (a_1 - u_1 x)/y$ and $v_2 = (a_2 - u_2 x)/y$. It is important to establish that the way a_1, a_2, u_1, u_2 are updated will ensure that v_1, v_2 will always have integer values.

Initially, $v_1 = (a_1 - u_1 x)/y = (x - x)/y = 0$ and $v_2 = (a_2 - u_2 x)/y = (y - 0)/y = 1$ are integers. Assume that $a_1 > a_2$ in some iteration (the other case can be symmetrically handled). At the beginning of the loop, we have integer values for v_1, v_2 such that Invariance (2) holds, that is,

$$a_1 = u_1 x + v_1 y$$
$$a_2 = u_2 x + v_2 y$$

But then,

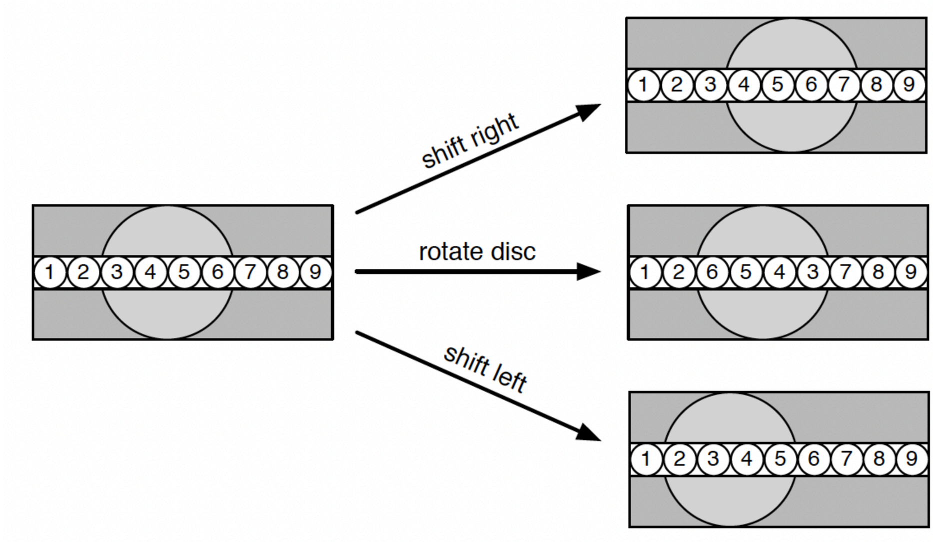
$$a_1 - a_2 = (u_1 - u_2)x + (v_1 - v_2)y$$

We update a_1 to $a_1 - a_2$, and u_1 to $u_1 - u_2$ in this case ($a_1 > a_2$). If we also update v_1 to $v_1 - v_2$ (and keep v_2 unchanged), then v_1, v_2 will continue to remain integers, and Invariance (2) will continue to hold.
The loop must terminate, because $max(a_1, a_2)$ strictly decreases in each iteration. At the end of the loop, we have $a_1 = a_2$, so by Invariance (1), $gcd(x, y) = gcd(a_1, a_2) = a_1 = a_2$. Moreover, by Invariance (2), we have $gcd(x, y) = a_1 = u_1 x + v_1 y$ for some integer v_1. This v_1 is computed as $(a_1 - u_1 x)/y$.

3. Consider a track holding 9 circular tiles. In the middle is a disc that can slide left / slide right and rotate 180◦. Thus, there are three ways to manipulate the current state:

    Shift Right: The center disc is moved one unit to the right (if there is space)
    Rotate Disc: The four tiles in the center disc are reversed
    Shift Left: The center disc is moved one unit to the left (if there is space)

The figure below illustrates these three manipulations.



Prove that if the track starts in an initial state with all but tiles 1 and 2 in their natural order, it is impossible to reach the goal state where all the tiles are in their natural order. These states are shown below.

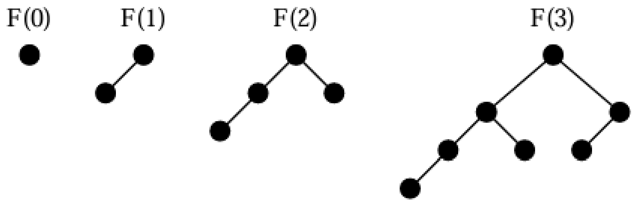
Solution: Let us define an **inversion** to be a pair of tiles that is out of their natural order. $\{Initial\ state:\ 1\ inversion\quad ..\ 2\ before\ 1\}\ \{Final\ state:\ 0\ inversions\}$

Loop invariance: Any transition does not change the parity of the number of inversions.

> If the transition is shift left or shift right : the left-to-right order of the tiles does not change and number of inversions remains the same - parity is same.
>
> If the transition is rotate disc : all pairs of tiles within the disc change their order = $\binom{4}{2} = 6$ pairs change their inversions. So, parity of the inversions remains the same.

Thus, any reachable state will have an odd number of inversions. So, goal state is unreachable.

4. Fibonacci trees F(n) are recursively defined for n ⩾ 0 as follows. F(0) is a single node. F(1) has two nodes with the root having only a left child. For n ⩾ 2, the tree F(n) consists of a root. Its left subtree is $F(n-1)$, and its right subtree is $F(n-2)$. The following figure shows the first four Fibonacci trees.



Prove the following assertions for all n ⩾ 0.
(a) F(n) contains $F_{n+3} - 1$ nodes (where F_i is the i-th Fibonacci number).
(b) F(n) contains F_{n+1} leaf nodes.
(c) The height of F(n) is n.
(d) At every non-leaf node v of F(n), the height of the left subtree of v is one more than the height of the right subtree of v. (Note that the empty tree has height −1, and a single-node tree has height 0.)

Solution:
(a) Prove by induction. Consider f(n) to be the number of nodes in F(n). Now, F(n) contains f(n-1) nodes in the left subtree, f(n-2) nodes in the right subtree and 1 root node. In all,

$$f(n) = f(n - 1) + f(n - 2) + 1$$

(b) Prove by induction. Consider l(n) to be the number of leaf nodes in F(n). Now, F(n) contains l(n-1) leaf nodes in the left subtree and l(n-2) leaf nodes in the right subtree.

$$l(n) = l(n - 1) + l(n - 2)$$

(c) Prove by induction. Consider h(n) to be the height of F(n). Now, F(n)'s height would be the same as the height of F(n-1) + 1 (edge between root node and left subtree).

$$h(n) = h(n - 1) + 1$$

(d) Notice that the subtree of any non-leaf node is a Fibonacci tree in itself. For some non-leaf node, let the left subtree be F(k) and the right subtree be F(k-1). As described in (c), h(k) = h(k-1) + 1.

5. Give a recursive definition of each of these sets of ordered pairs of positive integers.Use structural induction to prove that the recursive definition you found is correct. [Hint: To find a recursive definition, plot the points in the set in the plane and look for patterns.]
   a. S = {(a, b) | a ∈ Z^+, b ∈ Z^+, and a + b is even}
   b. S = {(a, b) | a ∈ Z^+, b ∈ Z^+, and a or b is odd}
   c. S = {(a, b) | a ∈ Z^+, b ∈ Z^+, and a+b is odd, 3 | b}

Solution:

a) Since we are working with positive integers, the smallest pair in which the sum of the coordinates is even is $(1,1)$. So our basis step is $(1,1) \in S$. If we start with a point for which the sum of the coordinates is even and want to maintain this parity, then we can add 2 to the first coordinate, or add 2 to the second coordinate, or add 1 to each coordinate. Thus our recursive step is that if $(a,b) \in S$, then $(a+2,b) \in S$, $(a,b+2) \in S$, and $(a+1,b+1) \in S$. To prove that our definition works, we note first that $(1,1)$ has an even sum of coordinates, and if (a,b) has an even sum of coordinates, then so do $(a+2,b)$, $(a,b+2)$, and $(a+1,b+1)$, since we added 2 to the sum of the coordinates in each case. Conversely, we must show that if $a+b$ is even, then $(a,b) \in S$ by our definition. We do this by induction on the sum of the coordinates. If the sum is 2, then $(a,b)=(1,1)$, and the basis step put (a,b) into S. Otherwise the sum is at least 4, and at least one of $(a-2,b)$, $(a,b-2)$, and $(a-1,b-1)$ must have positive integer coordinates whose sum is an even number smaller than $a+b$, and therefore must be in S by our definition. Then one application of the recursive step shows that $(a,b) \in S$ by our definition.

b) Since we are working with positive integers, the smallest pairs in which there is an odd coordinate are $(1,1)$, $(1,2)$, and $(2,1)$. So our basis step is that these three points are in S. If we start with a point for which a coordinate is odd and want to maintain this parity, then we can add 2 to that coordinate. Thus our recursive step is that if $(a,b) \in S$, then $(a+2,b) \in S$ and $(a,b+2) \in S$. To prove that our definition works, we note first that $(1,1)$, $(1,2)$, and $(2,1)$ all have an odd coordinate, and if (a,b) has an odd coordinate, then so do $(a+2,b)$ and $(a,b+2)$, since adding 2 does not change the parity. Conversely (and this is the harder part), we must show that if (a,b) has at least one odd coordinate, then $(a,b) \in S$ by our definition. We do this by induction on the sum of the coordinates. If $(a,b)=(1,1)$ or $(a,b)=(1,2)$ or $(a,b)=(2,1)$, then the basis step put (a,b) into S. Otherwise either a or b is at least 3, so at least one of $(a-2,b)$ and $(a,b-2)$ must have positive integer coordinates whose sum is smaller than $a+b$, and therefore must be in S by our definition, since we haven't changed the parities. Then one application of the recursive step shows that $(a,b) \in S$ by our definition.

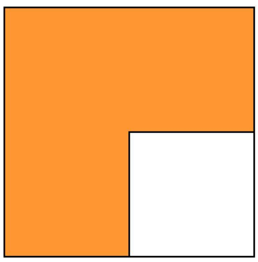
c) We use two basis steps here, $(1,6) \in S$ and $(2,3) \in S$. If we want to maintain the parity of $a+b$ and the fact that b is a multiple of 3, then we can add 2 to a (leaving b alone), or we can add 6 to b (leaving a alone). So our recursive step is that if $(a,b) \in S$, then $(a+2,b) \in S$ and $(a,b+6) \in S$. To prove that our definition works, we note first that $(1,6)$ and $(2,3)$ satisfy the condition, and if (a,b) satisfies the condition, then so do $(a+2,b)$ and $(a,b+6)$, since adding 2 or 6 does not change the parity of the sum, and adding 6 maintains divisibility by 3. Conversely (and this is the harder part), we must show that if (a,b) satisfies the condition, then $(a,b) \in S$ by our definition. We do this by induction on the sum of the coordinates. The smallest sums of coordinates satisfying the condition are 5 and 7, and the only points are $(1,6)$, which the basis step put into S, $(2,3)$, which the basis step put into S, and $(4,3)=(2+2,3)$, which is in S by one application of our recursive definition. For a sum greater than 7, either $a \geq 3$, or $a \leq 2$ and $b \geq 9$ (since $2+6$ is not odd). This implies that either $(a-2,b)$ or $(a,b-6)$ must have positive integer coordinates whose sum is smaller than $a+b$ and satisfy the condition for being in S, and hence are in S by our definition. Then one application of the recursive step shows that $(a,b) \in S$ by our definition.

6. Let S be a set defined recursively by
   Basis step: 1 ∈ S.
   Recursive step: For a positive integer k, if any element of the set {k, 2k+1, 3k} belongs to S, then all belong to S.
   For example, if 6 ∈ S, then 2 ∈ S, 5 ∈ S, 13 ∈ S and 18 ∈ S. Does 2023 ∈ S?
   Solution: From $\{k,\ 2k+1,\ 3k\}$, we get,
   $$k \Leftrightarrow 3k \Leftrightarrow 6k+1 \Leftrightarrow 12k+3 \Leftrightarrow 4k+1 \Leftrightarrow 2k \in S$$
   This shows that, $k \in S$ if and only if $2k$ and $2k+1 \in S$, or in the other way, $k \in S$ if and only if $\lfloor \frac{k}{2} \rfloor \in S$. Recursively, we get
   $$2023 \in S \Leftrightarrow 1011 \in S \Leftrightarrow 505 \in S \Leftrightarrow 252 \in S \Leftrightarrow 126 \in S \Leftrightarrow 63 \in S \Leftrightarrow 31 \in S$$
   $$\Leftrightarrow 15 \in S \Leftrightarrow 7 \in S \Leftrightarrow 3 \in S \Leftrightarrow 1 \in S$$

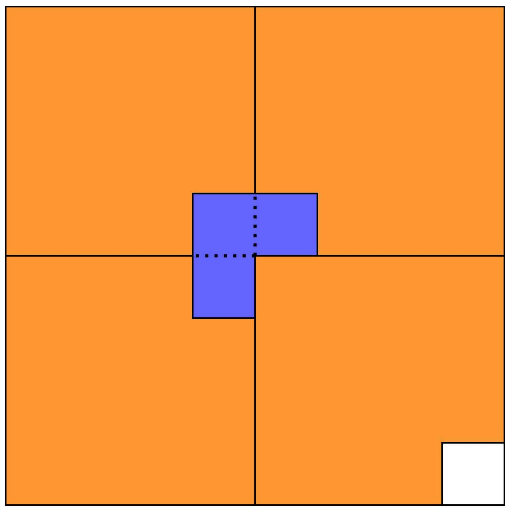
7. The positive integers 1,2,...,n are arranged in a random order. In one operation, two integers are interchanged. Prove that the initial ordering can never be reached after an odd number of operations.
   Solution: We track the number of "inversions": pairs {i, j} that occur in the wrong order in the list (if i < j and i is after j in the list, or vice versa). Swapping two elements of the list adds an odd number to the number of inversions. If we swap a and b, then:
   - If x is before both of them in the list, or after both of them, then the number of inversions with x is unaffected.
   - If x is between them in the list, then both pairs {x, a} and {x, b} change their state of being an inversion or not.
   - Finally, {a, b} changes its state.

   After an odd number of swaps, the number of inversions has different parity than the start, if the initial number of inversions are odd then final is even and vise-versa. so we cannot be in the initial state.

8. Give a recursive algorithm for tiling a $2^n x\ 2^n$ checker board with one square missing using right triominoes. Right triominoes are L-shaped tiles made up of three squares that cover exactly 1 unit square of the board.
   Solution: We follow a divide-and-conquer approach to tile the board.



n = 1 (Base case)



Construction for induction case

Basis: If n=1, then simply place the one right triomino so that its armpit corresponds to the hole in the 2×2 board.

Recursion: If $n > 1$, then divide the $2^n x 2^n$ board into 4 boards, each of size $2^{n-1} x 2^{n-1}$ (as shown in the diagram above). Note which quarter the hole occurs in, position one right triomino at the center of the board with its armpit in the quarter where the missing square is, and invoke the algorithm recursively four times-once on each of the $2^{n-1} x 2^{n-1}$ boards, each of which has one square missing (either because it was missing to begin with, or because it is covered by the central triomino).