**Homework 1: Basic Data Structures and Trees**

Handed out Thursday, Sep 5. Due at **11:59pm, Friday, Sep 13**. (See submission instructions below.)

**Problem 1.** (a) Consider the rooted tree of Fig. 1(a). Draw a figure showing its representation in the "first-child/next-sibling" form.

(b) Consider the rooted tree of Fig. 1(b) represented in the "first-child/next-sibling" form. Draw a figure showing the equivalent rooted tree.

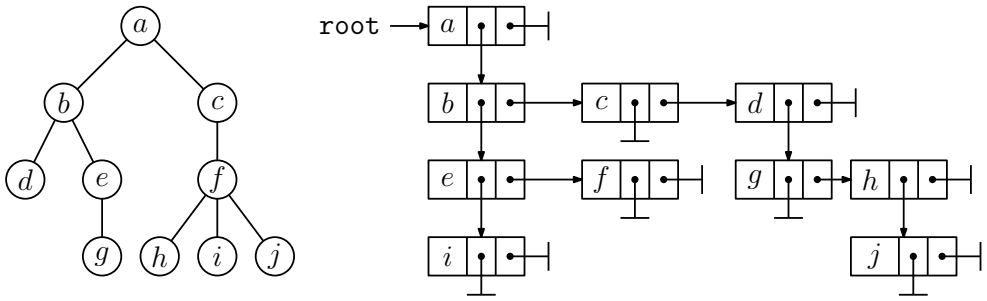


Figure 1: Rooted tree to first-child/next-sibling form and vice versa.

**Problem 2.** Consider the binary tree shown in Fig. 2. Draw a figure showing the inorder threads (analogous to Fig. 6 in Lecture 3).

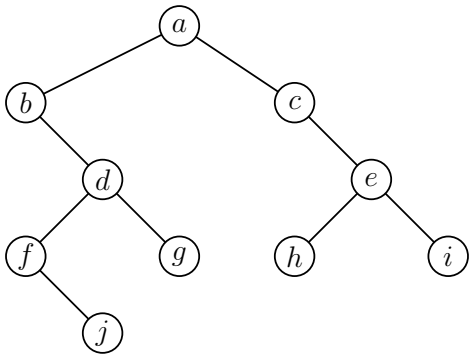


Figure 2: Add inorder threads to this tree.

**Problem 3.** A NASA scientist is running a numerical program on a large $n \times n$ matrix. She notices that the matrices she is working with have a large number of zero entries. In particular, the matrix entry $M[i][j]$ is definitely zero if either $i$ or $j$ is an even number (see Fig. 3). To save

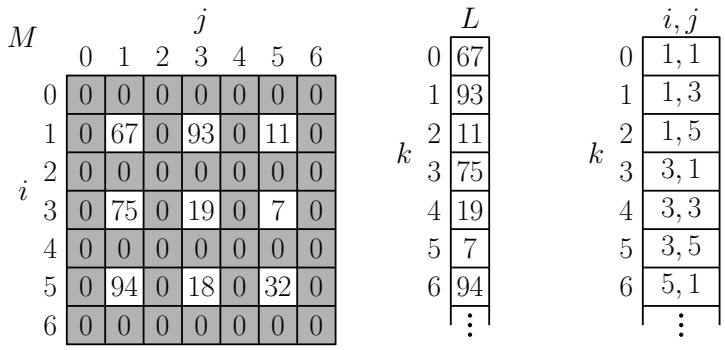


Figure 3: Mapping a matrix with many zeros to an array.

space, she decides to map the matrix entries to a smaller array, $L$, eliminating the entries that are known to be zeroes, as shown in Fig. 3.

(a) Give a function `int getValue(int i, int j, int n)`, which given $i$, $j$, and $n$ uses the array $L$ to retrieve the value of $M[i][j]$. (In the example above `getValue(4, 3, 7)` returns 0, and `getValue(5, 1, 7)` returns $L[6] = 94$.) If either $i$ or $j$ is even, the function should return zero. Otherwise, it should compute the index $k$ such that $L[k]$ corresponds to the entry $M[i][j]$, and return the value of $L[k]$. Your function should be correct irrespective of whether $n$ is even and odd, and you may assume that $n \geq 2$. For the sake of uniformity, you should assume that the entries of $M$ are mapped to the entries of $L$ as shown in the figure.

Another NASA scientist working a few doors down the hall has exactly the reverse situation. In her array, a matrix entry $M[i][j]$ is definitely zero if both $i$ and $j$ are odd numbers (see Fig. 4).

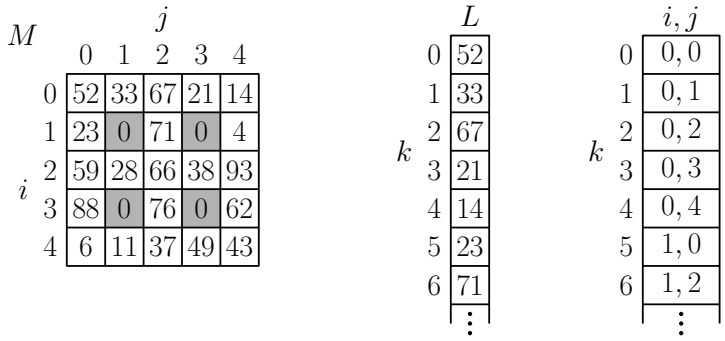


Figure 4: Mapping a matrix with not quite as many zeros to an array.

(b) Repeat (a) for this reversed situation. (Hint: This is trickier. It may help to consider two cases, one when $i$ is even and one when $i$ is odd.)

**Problem 4.** Suppose that you have an (unbalanced) binary search tree with inorder threads. Each

node has a left and right pointer, but no parent pointer. The objective of this problem is to show that you can use threads to locate the parent of each node.

Present pseudocode for a function `BinaryNode parent(BinaryNode p)` that returns a reference to `p`'s parent in the tree. You may assume that `p` is not `null`. If `p` is the root, this function should return `null`. The running time should be proportional to the tree's height.

As in class, you may assume that the binary-node pointer objects have been configured so that `p.left.isThread` indicates whether `p`'s left child link is a thread (and similarly for `p.right.isThread`). You may also assume that you can compare node references for equality. For example, given two objects `p` and `q` of type `BinaryNode` (where `q` is not `null`), you can test whether `p` is the left child of `q` with "`p == q.left`".

**Hint:** Be sure that your function works correctly when `p` is the leftmost or rightmost leaf of the tree. It is easy to get these cases wrong if you are not careful.

**General note regarding coding in homeworks:** When asked to present an algorithm or data structure, do **not** give complete Java code. Instead give a short, clean pseudocode description containing only the functionally important elements, along with an English description and a short example.

**Submission Instructions:** Because we will use GradeScope for grading, please write your solutions on the solution template, which can be downloaded from the class handouts page. (If there is insufficient space to write your answer, please note this within the space provided.)

Please submit your completed assignment through ELMS as a pdf file. If you submit through ELMS, you can either type your answer in some word-processing software (Word or LaTeX) and convert to a pdf document, or you can write your answer by hand and submit a scanned version. If you do the latter, please *do not* just submit the raw scanned image. Instead, use an image-enhancing app like CamScanner. (Poorly homeworks or illegible homeworks will be returned to you, and this may result in loss of points.)