Name: ______________________________________

1. (1 pt.)
   - **Read all material carefully.**
   - Budget your time: 35 minutes, 35 pts. $\Rightarrow$ 1 min./pt. avg.
   - You may refer to your books, papers, and notes during this test.
   - No computer or network access of any kind is allowed (or needed).
   - Write, and draw, carefully. Ambiguous or cryptic answers receive zero credit.
   - Use the conventions used in class and the textbook for notation, algorithmic options, etc.
   - There is one extra-credit question at the end. It is marked with a $\star$ and is harder (and graded more strictly) than the rest.

   Write your name in the space provided above.

2. (9 pts.) Determine the exact (not asymptotic) number of times the statement on line 5 (with comment *count me*) in the following code is executed.

   Express your answer as a function (as concise and simple as possible) of n and justify it briefly. Provide the exact value if $n = 23$.

```
1  int bogo = 0;
2  for(int i = 0; i < n; i++) { // see above for 'n'
3      for(int j = 0; j < i; j += 2) {
4          for(int k = 0; k < 5; k++) {
5              bogo = bogo + i * j; /* count me */
6          }
7      }
8  }
```

3. (10 pts.) Depict the *AVL tree* resulting from the insertion of the following keys, in the order listed, into an empty tree.

$$50, 70, 60, 90, 80, 65, 63, 62, 61, 10$$

   Show all intermediate steps. In particular, depict the state of the tree immediately following each insertion, before and after any necessary balancing operations. Identify the type of each balancing operation used and the root of the subtree to which it is applied.

[additional space for answering the earlier question]

4. (5 pts.) Depict the AVL trees resulting from the deletion of each of the following keys, in order, from the last tree of Question 3.

$$65, 61, 63$$

5. (10 pts.) We represent the empty binary tree by $\perp$ and a nonempty binary tree with root label n, left subtree l, and right subtree r by the triple (n, l, r). Consider the following function f on binary trees:

$$f(t) = \begin{cases} (n, \perp, \perp) & \text{if } t = (n, \perp, \perp) \\ (n, \perp, f(l)) & \text{if } t = (n, l, \perp) \text{ and } l \neq \perp \\ (n, f(r), \perp) & \text{if } t = (n, \perp, r) \text{ and } r \neq \perp \\ (n, f(l), f(r)) & \text{if } t = (n, l, r) \text{ and } l, r \neq \perp \\ \perp & \text{otherwise} \end{cases}$$

Depict, using the usual graphical conventions, the binary tree $f(T)$ where T is the final tree produced by the insertions in Question 3.

6. $\star$ (5 pts.) We use the notation $f^k(t)$ (with $k > 0$) to denote k nested applications of the function f, that is, $f(f(f(\ldots f(t))))$, where there are k instances of f in the expression.

   Using the definitions of f and T from Question 5, depict, using the usual graphical conventions, the binary trees $f^{20}(T)$ and $f^{21}(T)$. *Explain your answers.* (There is no credit for answers without proper explanations.)