15-451 Algorithms, Spring 2012

Homework 3 (100 pts) — Due: Feb. 15-18

oral presentation

Ground rules:

- This is an oral presentation assignment. You should work in groups of **exactly** three. At some point your group should sign up for a 1-hour time slot on the signup sheet on the course web page. By signing up with fewer than three group members you are giving course staff permission to place unpaired students in your group.
- Each person in the group must be able to present every problem. The TA/Professor will select who presents which problem. The other group members may assist the presenter.
- You are not required to hand anything in at your presentation, but you may if you choose. If you do hand something in, it will be taken into consideration (in a non-negative way) in the grading.

Question	Points	Score
1	30	
2	35	
3	35	
Total:	100	

1. **Modern Rosetta Stone.**

For half of a century, we have been broadcasting news, music and television shows to outer space. Last month, an odd interference was found in the electromagnetic spectrum. After weeks of investigation, scientists discovered that the message was a modern-day Rosetta Stone sent from an extraterrestrial society. The US government has solicited you for your computer science expertise to aid in communicating with this society.

When deciphered, the message indicates the relationship between the set of English words and the set of alien words, Σ. A major barrier for translation is that the words in Σ are highly ambiguous when left out of context and the sentences are extremely complex. Luckily, linguists have determined that the rules for forming sentences in the alien language is closely related to English, which are provided as follows:

Sentence	=	[Noun-Phrase] + [Verb-Phrase]
Verb-Phrase	=	[Verb-Phrase] + [Prepositional-Phrase]
Verb-Phrase	=	[Verb] + [Noun-Phrase]
Prepositional-Phrase	=	[Preposition] + [Noun-Phrase]
Noun-Phrase	=	[Determiner] + [Noun]
Noun		
Verb		
Preposition		
Determiner		

Other computer scientists have already developed data structures which allow you to query for any $\sigma \in \Sigma$ if σ is a noun, verb, preposition, or determiner in constant time. Your task is to write an $O(n^3)$ dynamic programming solution to verify sentences and give the set of sentence structures which match the given string. First, you will write an algorithm which returns true iff the input string represents a valid sentence.

(5) (a) Identify the structure of the table (bottom-up) or function (top-down) by describing the cell contents or the return value, respectively.

(5) (b) Define the base case for your algorithm.

(5) (c) Give the recursive definition of your algorithm.

(5) (d) Prove correctness of your algorithm.

(5) (e) Prove your algorithm runs in $O(n^3)$ time.

(5) (f) Provide an algorithm for recovering all possible sentence structures given a string which has been verified.

2. **Self-Adjusting Tree for Strings**

In this problem we will develop a self-adjusting search tree to store strings. Suppose that Σ is a finite alphabet with a natural linear order.

The string will be stored in a ternary tree. Thus each node may have three children; Left, middle, and right which may be empty. Some of the node may also be marked as terminal characters of some stored string.

In Figure 2 we give such a ternary search tree to store English words. The square boxes are the terminal characters. As an example to find the word **door** in the tree. we first search for the character **d** in the root. Since **b** is less than **d** we search the right subtree, finding **d**. We now search the middle tree of **d** looking for **o**. We find the **o** in the right subtree of **a**. Continuing in this fashion we finally find the **r** in the right most element in the tree.

(5) (a) Explain how one can define splay rotations for our ternary search tree such that the words stored in the tree do not change.

(10) (b) Give algorithms for search and insert a string in our ternary tree.

(10) (c) Define and explain a potential function for search and insert operations.

(10) (d) State and prove an Access-Lemma.

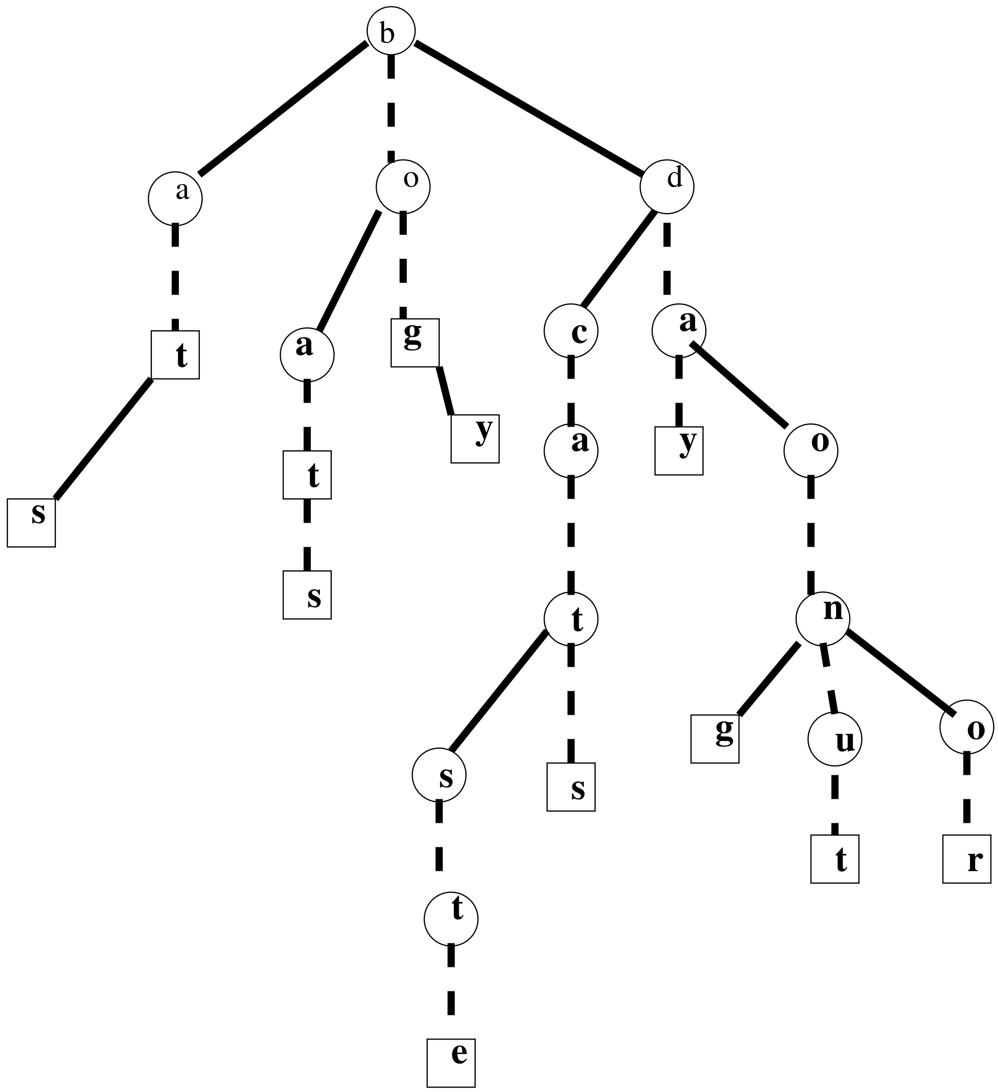

Figure 1: A search tree for the words: **at, as, bat, bats, bog, boy, caste, cats, day, dog, donut, door.**

(15) 3. **Treaps.**

Given a treap with n elements. As in lecture, we identify elements by their keys and priorities.

(5) (a) What is the expected number of leaves in that treap?

(5) (b) What is the probability that an element with j priority is a common ancestor of elements with i and k priorities? Assume that $1 \leq i \leq j \leq k \leq n$.

(10) (c) What is the expected length of the unique path from the elements with priorities i and k defined in the previous part?