Goals. This assignment's purpose is to provide a hands-on introduction to the latent structure underlying how words are arranged in language—that is, syntax. This assignment focuses on constituencies, dependencies, parsing, and parts of speech.

Part 1: Sequence Labeling

Take the following transition probabilities and emission probabilities:

```
Transition probabilities:
P(D | START) = 0.5  P(N | START) = 0.4  P(V | START) = 0.1
P(N | D) = 0.9      P(V | D) = 0.1
P(V | N) = 0.6      P(N | N) = 0.3      P(D | N) = 0.1
P(N | V) = 0.4      P(D | V) = 0.3      P(V | V) = 0.3

Emission probabilities:
P("the" | D) = 1.0      P("dog" | N) = 0.3
P("fish" | V) = 0.4     P("can" | V) = 0.4
P("fish" | N) = 0.2     P("can" | N) = 0.2
```

Q1 (10 points). Use the Viterbi algorithm to find the most likely tag sequence for the sentence "the fish can fish". Do this by filling in each valid cell of the Viterbi chart (8 points). If a word cannot be associated with a given POS tag, you do not have to fill in that cell. After drawing the chart, provide the most likely tag sequence (2 points).

	the	fish	can	fish
D				
N				
V				

Q2 (3 points). In 2–3 sentences, why might an HMM tagger trained on news articles perform poorly on text from social media?

Part 2: Constituencies and Context-free Grammars

As we've discussed in class, words can often work together to form higher-order structures called **constituencies**. Constituencies can recurse into ever-higher-order structures, giving language a sort of infinite potential complexity. While natural language is technically not context-free,[1] context-free grammars (CFGs) still provide a nice theoretical playground for us to work through how to think about these latent structures.

Q3 (5 points). Here are three constituency trees that were each generated by the same CFG:
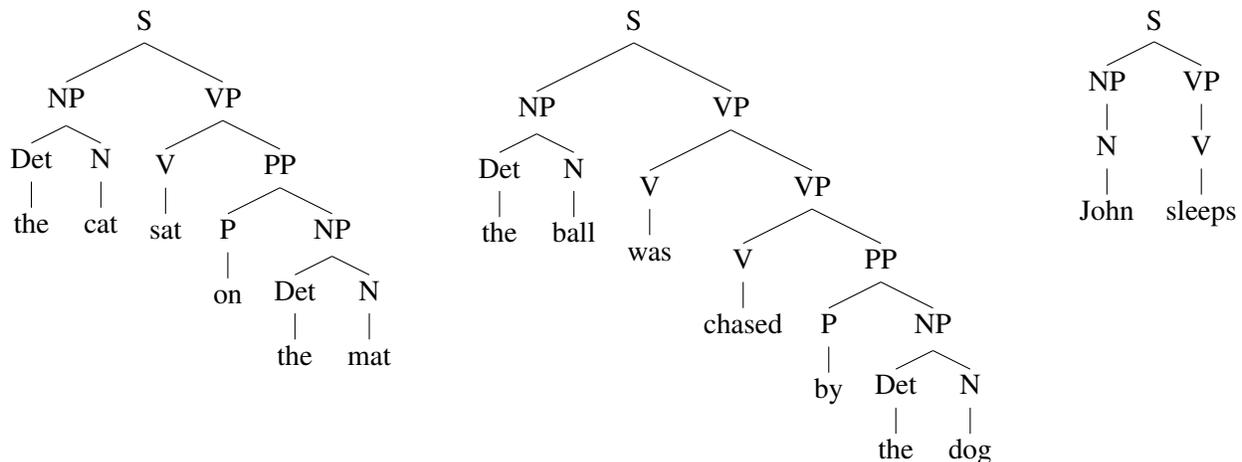
Write the minimal CFG that could generate each of these sentences.

Q4 (6 points). The following is a probabilistic context-free grammar (PCFG). The brackets to the left of each rule correspond to the probability that the given nonterminal will generate the associated right-hand side; the probabilities *per non-terminal* sum to 1.

```
[0.6] S -> A B
[0.4] S -> A C
[1.0] A -> x
[0.5] B -> y
[0.5] B -> z
[0.5] C -> y
[0.5] C -> z
```

How many parses are there for the string "x y"? Which parse is the most probable under the provided grammar? Provide each parse alongside its probability; feel free to either draw constituency trees or use bracket notation.

Q5. The sentence "The girl saw the boy with a telescope" is ambiguous. Here, you will show precisely how it is ambiguous.

(a; 6 points) Provide constituency trees for both possible parses, and state which of the parses you believe is more probable. Use only the following non-terminals: S, NP, VP, PP, N, V, P, Det.

---

[1]https://www.eecs.harvard.edu/shieber/Biblio/Papers/shieber85.pdf

(**b; 4 points**) Based on your answer, write a minimal probabilistic CFG that yields both parses, and assigns a higher probability to the more probable parse.

Q6.   In class, we've walked through the CKY algorithm using an upper-triangle matrix, like the following:

she   ate   rice   with   chopsticks

Here is a PCFG for this sentence:

```
[1.0] S  -> NP VP              [1.0] V -> ate
[0.3] NP -> she                [1.0] P -> with
[0.2] NP -> rice
[0.2] NP -> chopsticks         [0.6] VP -> V NP
[0.3] NP -> NP PP              [0.4] VP -> VP PP
[1.0] PP -> P NP
```

(**a; 10 points**). Fill in the CKY matrix using the provided PCFG. Include the fully filled out matrix in your written report. Be sure to include probabilities for each non-terminal in each non-empty cell.

(**b; 2 points**). How many valid parses are there? Which parse is the most likely according to the given PCFG?

(**c; 2 points**). In your own words, describe the sentence's meaning according to each parse. Use only 1–2 sentences per possible parse.

Part 3: Dependencies

Unlike constituencies, dependencies do not presume that there are latent structures above the word level. Dependencies are asymmetric relationships between *heads* and their *dependents*. For example, a subject is a dependent of its verb head, and an adjective is a dependent of its noun head. These relationships are depicted as arrows from heads (parents) to dependents (children).

Q7. (5 points)   Draw the dependency tree for the following sentence: "Students who study syntax understand a parse." Label each arc with its dependency label. You should only need the following dependency labels: nsubj, dobj, relcl, det.

Q8. (10 points)   Consider the sentence "The cat chased mice." We'll use a shift-reduce parser to draw the dependency tree. Assume a transition system like the one we've covered in class with three operations: SHIFT, LEFT-ARC, RIGHT-ARC.

Starting with an empty stack and the full sentence in the buffer, show the sequence of operations needed to build the correct dependency tree using a shift-reduce parser. For each step, show the following: the stack contents, the buffer contents, the operation performed for a given step, and any dependency relations created in that step. Use only the following dependencies: det, nsubj, dobj. Note that you may not need all rows of this table.

Step	Stack	Buffer	Operation	New Relations
0	[ROOT]	[The, cat, chased, mice]	SHIFT	—
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Q9. Consider the sentence "Dogs that bark loudly scare children." Here is a constituency tree (top) and a dependency tree (bottom) for this sentence:
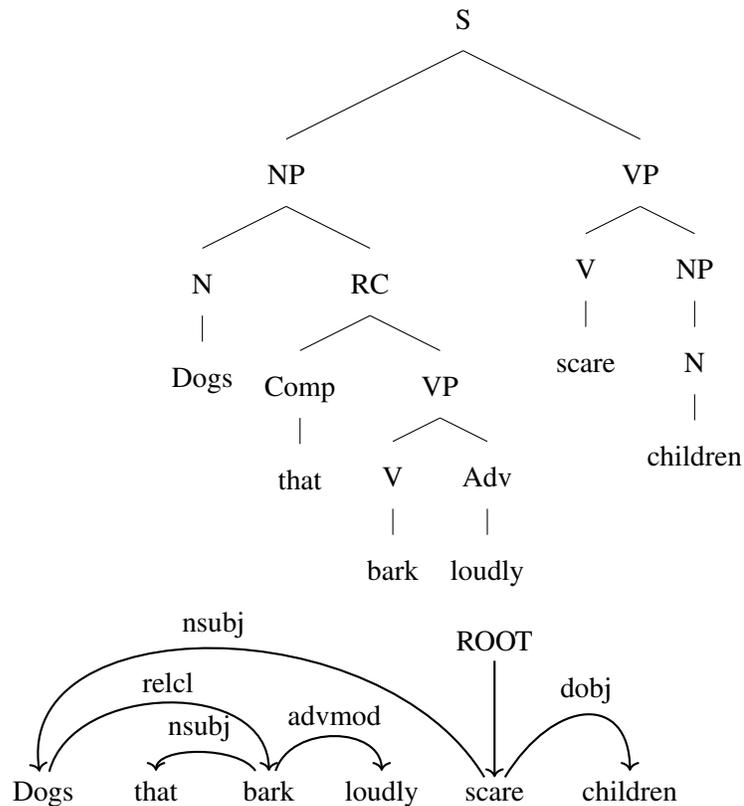
(a; 4 points) In the constituency tree, which node is the relative clause ("that bark loudly") a child of? In 1–2 sentences, what does this grouping imply about the relationship between the relative clause and the noun "Dogs"?

(b; 3 points) In the dependency tree, which word is the head (main word) of the relative clause? What word is this head a dependent of? What is the label of this head–dependent arc?

(c; 3 points) Consider the constituent (subtree) defined by the top-level NP node ("Dogs that bark loudly"). Using just "yes" or "no", does any single node or arc in the dependency tree capture this grouping? In 1–3 sentences, what specific information does this constituency subtree encode that is absent from the dependency tree?

Extra Credit

E1. (6 points)   As stated in Part 1, natural language is not actually context-free. Come up with an example sentence that shows that natural language is not context-free, and draw its dependency parse (just the arrows; you do not have to label the edges). You may not use the examples from the class slides, nor the example given in the linked paper by Stuart Shieber.

E2.   Consider the sentence "The horse raced past the barn fell." This is a grammatical English sentence (believe it or not), but it is difficult for people to parse.

(a; 3 points) Draw the constituency tree for this sentence.

(b; 3 points) Draw the constituency tree for "The horse raced past the barn."

(c; 5 points) In 2–3 sentences, explain why humans find "The horse raced past the barn fell." difficult to process, and what specific structural property of the sentence causes this difficulty.

Submission

Upload your write-up to Gradescope as a PDF to **HW3: Syntax and Parsing**.