

FIFTH QUIZ

You have 15 minutes from the start of class to complete this quiz. Read the questions with care; work with deliberate speed. Don't give us more than we ask for. The usual instructions apply. Good luck!

Problem 1 (11 points)

Below is a grammar for radio station names, in which terminal symbols are indicated by this typewriter typeface. The starting symbol is RadioStation; *space* indicates a space character and *empty* indicates the empty string.

RadioStation → CallLetters *space* Frequency

CallLetters → FirstLetter OtherLetters

FirstLetter → K | W

OtherLetters → Letter Letter | Letter Letter Letter

Letter → A | B | ... | Z

Frequency → AMFrequency | FMFrequency

AMFrequency → AMNumber *space* AM

AMNumber → Digit Digit Digit | Digit Digit Digit Digit

Digit → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

FMFrequency → FMNumber *space* FM

FMNumber → FMLeft FMRight

FMLeft → Digit Digit | Digit Digit Digit

FMRight → *empty* | . Digit

(a) (7 points) Mark each of the following eight station names “legal” or “illegal” according to this grammar.

KLUV 1000 AM

WUV 108 FM

LOVE 89.5 FM

KHRT 108.0 AM

WVAL 007 AM

WOVE 1250

KUPID 101.0 FM

KISS 88.1 FM

(b) (4 points) Choose one of the legal radio stations above and draw a derivation tree (parse tree) showing how to derive it from the grammar. You may abbreviate the nonterminal symbols in the rules; fitting your tree in the space might also be easier if you draw your tree from left to right instead of from top to bottom.

Problem 2 (14 points)

(a) (8 points) Draw a state transition diagram for an FSA that accepts the same valid AM radio stations that the grammar from Problem 1 generates; you don't have to accept any FM stations. You may omit drawing the error state. Be sure to indicate the initial state and the accept state(s). [Hint: Be careful to count the digits correctly.]

(b) (6 points) Draw the state transition table for the *first six states* of your FSA above. You may leave blank any unspecified transitions; you should omit the error state. We have supplied horizontal lines; you will supply the vertical lines and everything else.

Here are some topics we might have asked questions about on this quiz, if there had been enough time and space. Similar questions may appear on future quizzes or on the final.

- What was (or is) the “software crisis”? Describe each of the following and how they helped (or didn’t) to resolve the software crisis: structured programming, n-version programming, formal verification.
- How do stacks, queues, priority queues, and dequeues behave? Given a series of operations on one of these data structures, what’s the state of the structure after executing those operations?
- Write Java code to traverse some data structure (e.g., an ArrayList or a file) and (a) count the elements, or certain elements, (b) find the largest or smallest element, (c) “fold” the elements, or certain elements (e.g., add them, or collect them, or combine them in some other way), or (d) compute the average value of the elements, or certain elements.
- Given some Java code, identify which class, method, or lines perform some task, or which lines would have to be changed to produce some different behavior.
- What is the purpose of Java’s exception handling mechanism, when does a method need a “throws” clause, when should we use exceptions rather than in-line code to handle errors, what is the flow of control through a series of try and catch and finally clauses?
- Why do we care about natural language understanding and why haven’t we been able to achieve it yet?