Sample midterm - ANSWERS

Question 1

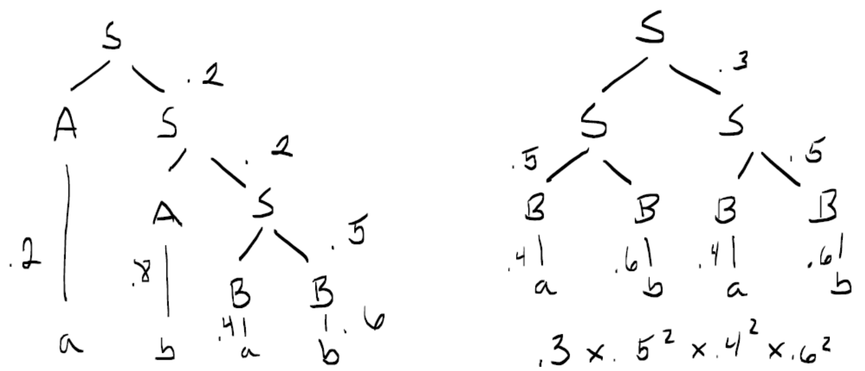
a) Snowball

b) Snowed

c) Snowy

Question 2

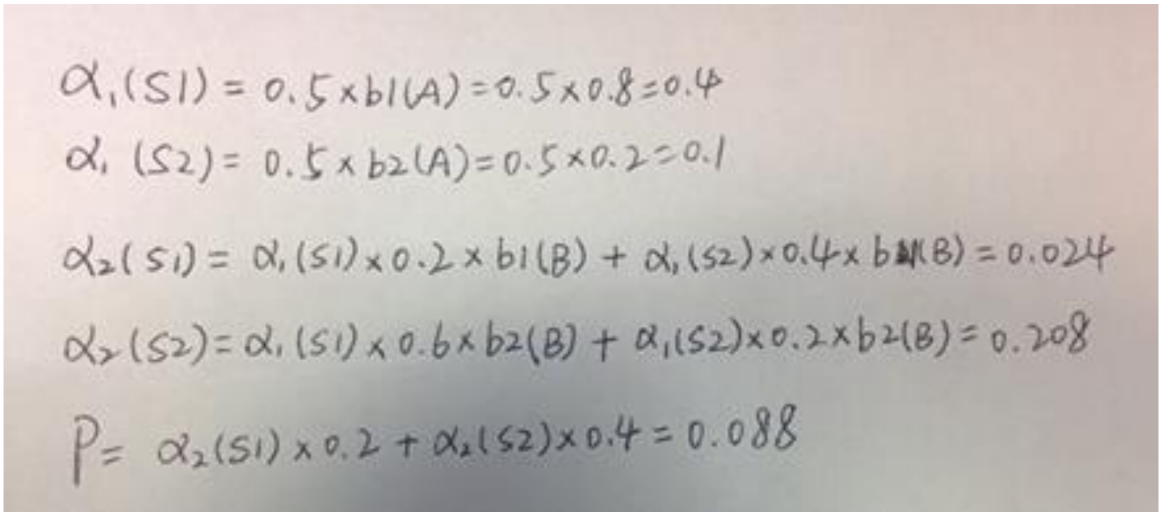
2)



$.2^3 \times .5 \times .8 \times .4 \times .6 = .0008$

$.3 \times .5^2 \times .4^2 \times .6^2 = .004$

Question 3



$\alpha_1(S1) = 0.5 \times b1(A) = 0.5 \times 0.8 = 0.4$

$\alpha_1(S2) = 0.5 \times b2(A) = 0.5 \times 0.2 = 0.1$

$\alpha_2(S1) = \alpha_1(S1) \times 0.2 \times b1(B) + \alpha_1(S2) \times 0.4 \times b2(B) = 0.024$

$\alpha_2(S2) = \alpha_1(S1) \times 0.6 \times b2(B) + \alpha_1(S2) \times 0.2 \times b2(B) = 0.208$

$P = \alpha_2(S1) \times 0.2 + \alpha_2(S2) \times 0.4 = 0.088$

Question 4

True, the probability of a sentence is equal to the sum of probabilities of all possible parse trees of this sentence. A possible parse tree is any tree that yields the sentence (sequence of tokens).

Question 5

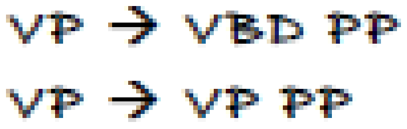
The if statement is the key step in building the table. Essentially, we are trying to determine the most probable parse tree for word sequence i...j. Therefore, we iterate over all possible rules in the grammar which have a nonzero probability for this sequence and split point. The if statement checks each split point and rule to see if they produce the highest probability tree for the sequence i...j. If so, we store that value.

This algorithm works from the bottom up and starts with smaller phrases, building off those until the most probably parse is determined for the whole tree. Until the most probable parse is determined for the whole tree.

Question 6

a) Viterbi

Question 7



VP → VBD PP
VP → VP PP

Question 8

One possible PCFG is:

S -> A	p=1
A -> the	p=.4
A -> A the	p=.6

Any sequence of n the’s requires 1x the first rule, n-1x the last rule, and 1x the middle rule.

Question 9.

VP

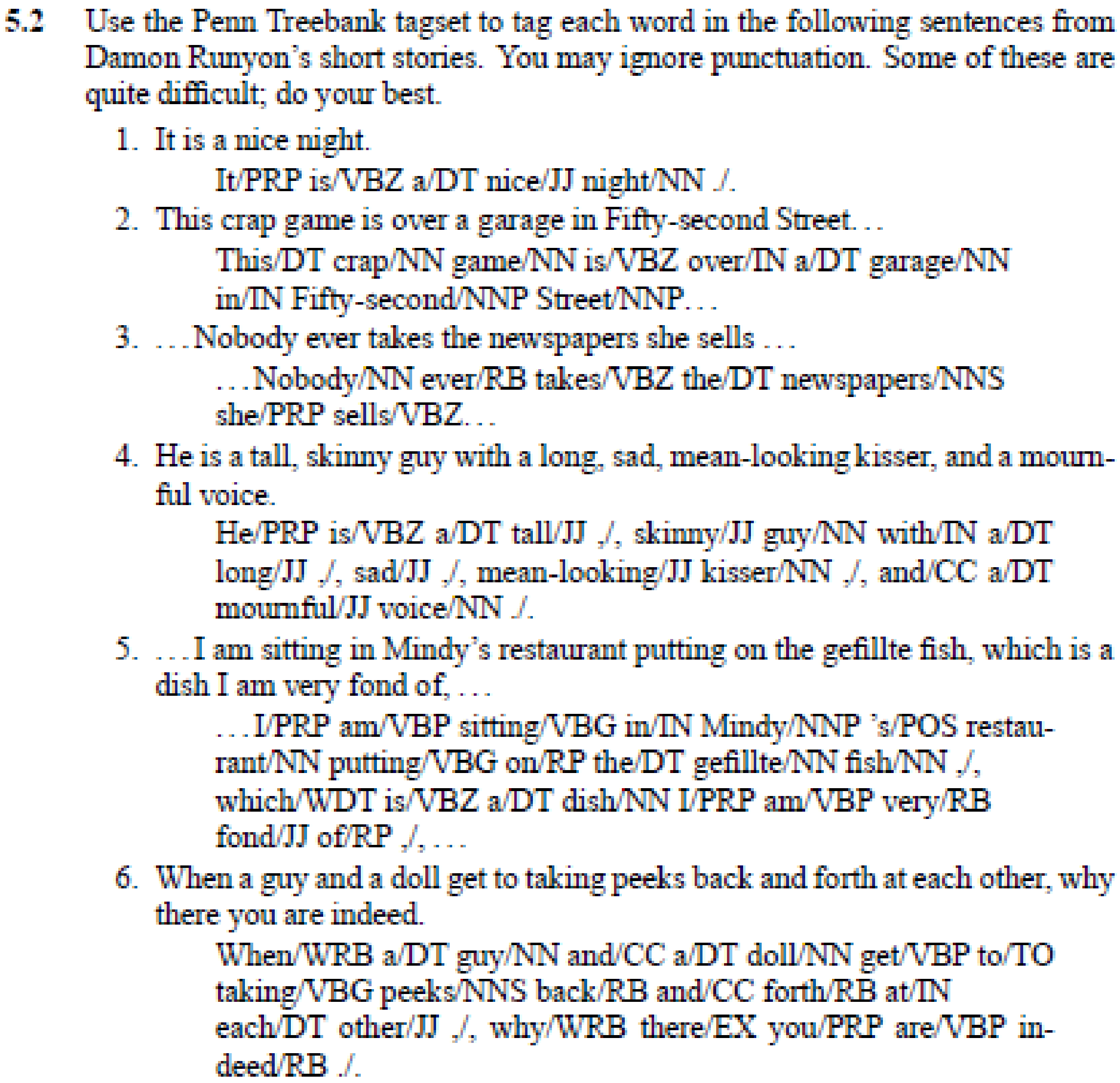
N

VP

V

IN

Question 10.



5.2 Use the Penn Treebank tagset to tag each word in the following sentences from Damon Runyon's short stories. You may ignore punctuation. Some of these are quite difficult; do your best.

1. It is a nice night.

   It/PRP is/VBZ a/DT nice/JJ night/NN ./.

2. This crap game is over a garage in Fifty-second Street. . .

   This/DT crap/NN game/NN is/VBZ over/IN a/DT garage/NN in/IN Fifty-second/NNP Street/NNP. . .

3. . . . Nobody ever takes the newspapers she sells . . .

   . . . Nobody/NN ever/RB takes/VBZ the/DT newspapers/NNS she/PRP sells/VBZ. . .

4. He is a tall, skinny guy with a long, sad, mean-looking kisser, and a mournful voice.

   He/PRP is/VBZ a/DT tall/JJ ,/, skinny/JJ guy/NN with/IN a/DT long/JJ ,/, sad/JJ ,/, mean-looking/JJ kisser/NN ,/, and/CC a/DT mournful/JJ voice/NN ./.

5. . . . I am sitting in Mindy's restaurant putting on the gefillte fish, which is a dish I am very fond of, . . .

   . . . I/PRP am/VBP sitting/VBG in/IN Mindy/NNP 's/POS restaurant/NN putting/VBG on/RP the/DT gefillte/NN fish/NN ,/, which/WDT is/VBZ a/DT dish/NN I/PRP am/VBP very/RB fond/JJ of/RP ,/, . . .

6. When a guy and a doll get to taking peeks back and forth at each other, why there you are indeed.

   When/WRB a/DT guy/NN and/CC a/DT doll/NN get/VBP to/TO taking/VBG peeks/NNS back/RB and/CC forth/RB at/IN each/DT other/JJ ,/, why/WRB there/EX you/PRP are/VBP indeed/RB ./.

Question 11.

(a) “The horse raced past the barn fell.” – This can be challenging because most NLP systems these days are statistical rather than rule-based. A rule-based system could likely rule out the incorrect parse here, but a statistical system could get confused here since “the horse raced past the barn” (without the last verb) is a likely sentence, and even with “fell” mistagged could be considered a more likely parse than the correct parse.

(b) “Omnēs enim quī accēperint gladium, gladiō perībunt.” (student-provided answer)– I chose a non-English sentence here because this is a much bigger problem in languages with more inflection than English (which has very little inflection left anymore). The sentence means roughly “For all who take up the sword will die by the sword”. The word for “take up” here is “accēperint”. The morphology of this verb is actually ambiguous. It could be either a perfect subjunctive, meaning “might have taken up the sword” or a future perfect “will have taken up the sword”. It's actually not even 100% clear to a human reader which of the two it is, much less so for a computer.

If we're considering this in the context of a machine translation system, before the system could even decide whether to translate “might have taken up” or “will have taken up”, it needs to know what verb it's looking at. “accēperint” is an inflected form of the verb “accipio”. But as it turns out, “accēperint” is a fairly uncommon inflection. (“might have” and “will have” are uncommon things to say) It's quite possible that the system will never have seen the word “accēperint” before (even if it's seen other perfect subjunctive or future perfects before), and if it didn't have a way to process morphology in a general way, it may not know that this is a form of “accipio”.

(c) “John and Jane like apples and oranges” – This sentence is ambiguous, even for a human. If you are trying to discover the semantics of this sentence, for example, it's not clear whether the correct answer should include “like(John, oranges)”

Question 12.

verb → noun : hack → hacker

adjective → noun : happy → happiness

adjective → adverb : slow → slowly

adjective → verb : little → belittle

Question 13.

S → NP VP	NNP → Joe
S → VP	NNP → Maureen
SQ → VBD NP VP	NN → dinner
S → WHADVP SQ	VBD → did
WHADVP → WRB	VBD → ate
NP → NNP	WRB → when
NP → NN	VB → eat
VP → VB NP	
VP → VBD NP	

Question 14.

1. NP – run a race

2. NP NP – give my friend an apple

3. Ø – sleep

4. P_{to} - want to eat.

5. S – said he was fired.

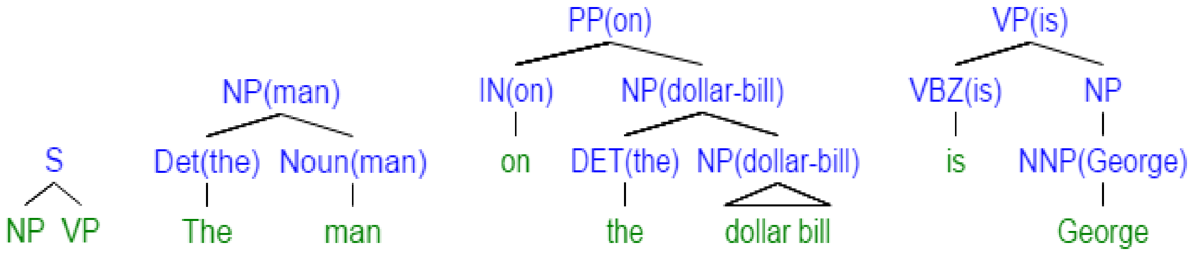
Question 15.

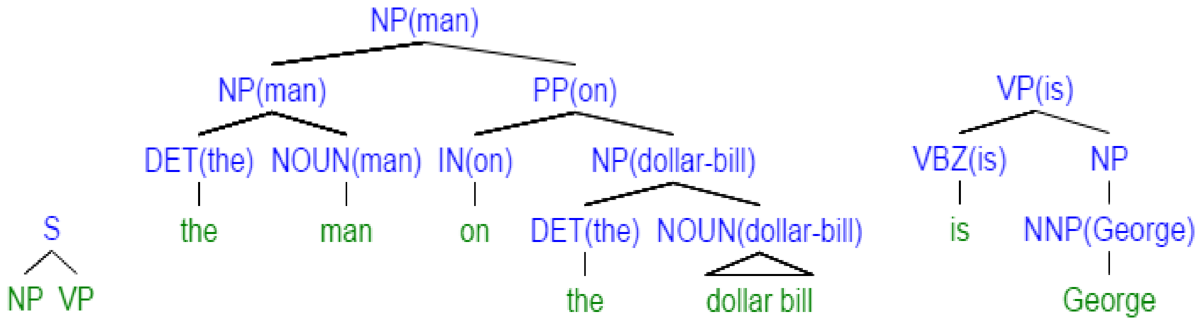
A transformation-based POS tagger is a mixture of a statistical and a rule-based tagger. It assigns initial POS to each token based on the most frequent POS for that word (this part is statistical), and then applies rule-based transformations to these POS trying to correct possible tagging errors.

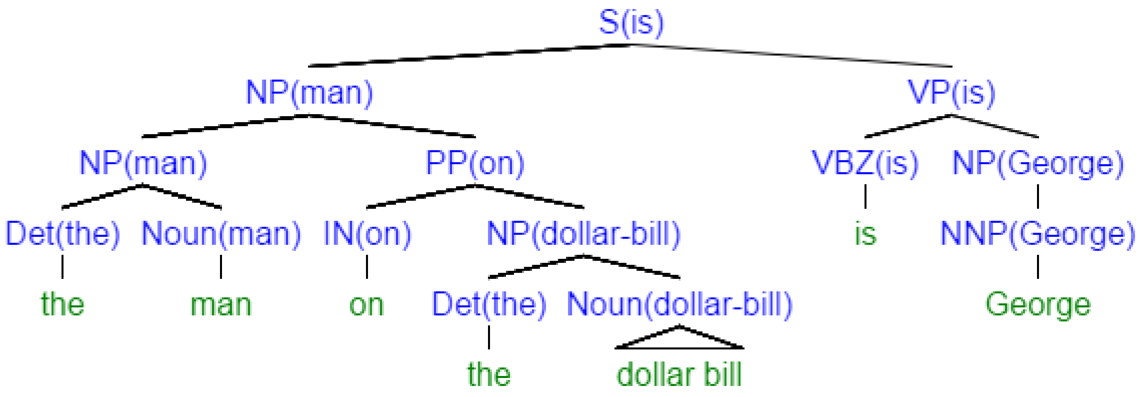
Question 16.

One way to deal with long-distance dependencies is to begin with smaller trees of various words in the sentence (that are the most probable according to statistical parsing), then put the smaller trees together to form the full tree. The heads of each non-terminal are also shown.

Ex: “The man on the dollar bill is George”







Question 17.

A → B X | a | f | g

B → b

C → c

D → d | e

X → C D

X → h

Where A, B, C, D, X are non-terminals, A is the start symbol, and a, b, c, d, e, f, g, h are terminals.

Other solutions exist.

Question 18.

O(n^3 x |G|) where n is the length of the parsed string and |G| is the size of the grammar (note: G in CNF)

Question 19

(S\NP)/NP

Question 20

(1)	(2)	(3)	(4)
a - b c d	A	ABC	B

Question 21

d

Question 22

c

Question 23

d

Question 24

(no answer provided)