START OF QUIZ
Student ID:
56633415,Cady,Amandisa

Question 1

Topic: Lecture 3
Source: Lecture 3

When we nest deep structures in dictionaries, we lose their O(1) benefits. Can you think of a better way to represent complex data sets? (1)

Question 2

Topic: Lecture 3
Source: Lecture 3

When would we want to represent linguistic data in a list, instead of a dictionary or a set? (1)

Question 3

Topic: Lecture 2
Source: Lecture 2

How does Zipf's law help explain the distribution of word frequencies in a corpus? What impacts does that have on our algorithms? (1)

Question 4

Topic: Lecture 1
Source: Lecture 1

Why is the .split() method useful when working with sentences or phrases? (1)

Question 5

Topic: Lecture 2
Source: Lecture 2

What role does linguistic annotation provide for corpora, specifically for computational linguistics? (1)

Question 6

Topic: Lecture 4
Source: Lecture 4

Attributive adverbs are a type of adverb that provides ”flavour” to speech verbs (example: ”she said quickly”; ”he spoke loudly”). They are often frowned upon in formal writing, because they can be replaced with other verbs: ”blurted” or ”shouted”, in the example. Write a quick function that finds them in the Brown corpus, and reports how many sentences in 1000 have them. (2)

Question 7

Topic: Lecture 4
Source: Lecture 4

In French, negation is often indicated by ”ne ... pas” (ie, ”je ne parle pas” - ”I am not speaking”; ”tu ne conduis pas” - ”You are not driving”, etc.). However, in speech, one of the two is often dropped: ”je ne parle.” or ”tu conduis pas.”. Using this information, how would you determine whether a corpus was composed of written or spoken French? You don't need to write the code, but explain the logic that you would use to come to this conclusion. (2)

Question 8

Topic: Lecture 1
Source: Lecture 1

Write a function that capitalizes the first letter of each word in a string, without using the .title() method or any external libraries. What are some assumptions that you are making? (2)

Question 9

Topic: Long
Source: Lecture 1

Write a function that validates if a string matches a phone number format, such as (123) 456-7890. What types of invalid inputs should the function check for? Are there edge cases we would be willing to accept? How would we handle those? Write 3 test cases - 2 that should pass, and one that should fail. (3)

END OF QUIZ