**Report 2:**  
Shilo Avital 206487407 Tamar Michelson 323805861

**RNN Acceptor Challenge Report - Part 2**

**1. Overview**

This report describes three languages (8–10) that challenge the RNN acceptor's ability to distinguish between valid and invalid sequences. Each case defines a language and its complement, aiming to find weaknesses in the LSTM-based model.

**2. Language 8**

\*\*Language Description:\*\*  
Each string begins with an integer value, followed by a random sequence of lowercase letters. The prefix integer equals the ASCII value of the most frequent character in the string plus its frequency count.

\*\*Challenge Rationale:\*\*  
This task requires counting occurrences and computing a non-linear transformation. It involves non-local dependencies and basic arithmetic, which RNNs struggle with.

\*\*Experiment Outcome:\*\*  
The model failed to learn this mapping correctly, showing low accuracy on both train and test sets.

***\*\*Plot:\*\* (Insert evaluation plot here)***

**3. Language 9**

\*\*Language Description:\*\*  
Strings that begin and end with the same character (e.g., 'abca', 'zxxxz').

\*\*Challenge Rationale:\*\*  
The dependency is between distant characters (first and last), which LSTMs find difficult to capture.

\*\*Experiment Outcome:\*\*  
The model did not generalize well, performing close to chance level. Accuracy plateaued early.

***\*\*Plot:\*\* (Insert evaluation plot here)***

**4. Language 10**

\*\*Language Description:\*\*  
Strings with even length are considered valid, while strings with odd length are invalid.

\*\*Challenge Rationale:\*\*  
This requires a global property of the sequence (its length), which cannot be easily inferred from local patterns.

\*\*Experiment Outcome:\*\*  
The model failed to detect this property consistently, likely because the decision is not based on token content.

***\*\*Plot:\*\* (Insert evaluation plot here)***

**5. Summary**

Languages 8–10 were designed to probe limitations of sequence-based models like LSTMs. All three languages resulted in model failure, reinforcing known difficulties in modeling counting, distant dependencies, and global properties using standard recurrent architectures.