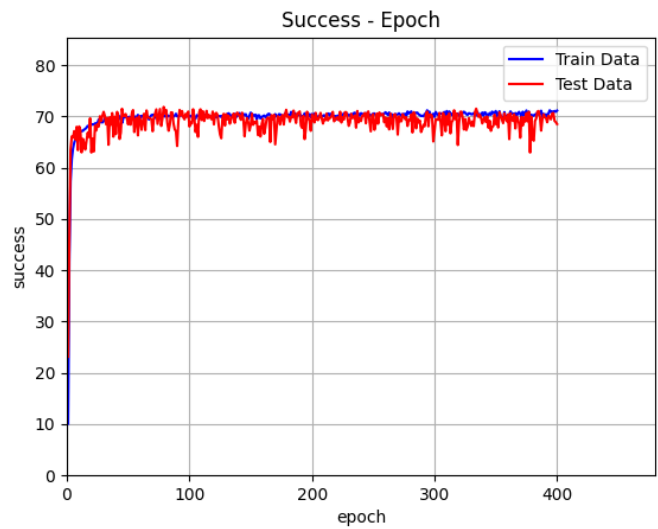
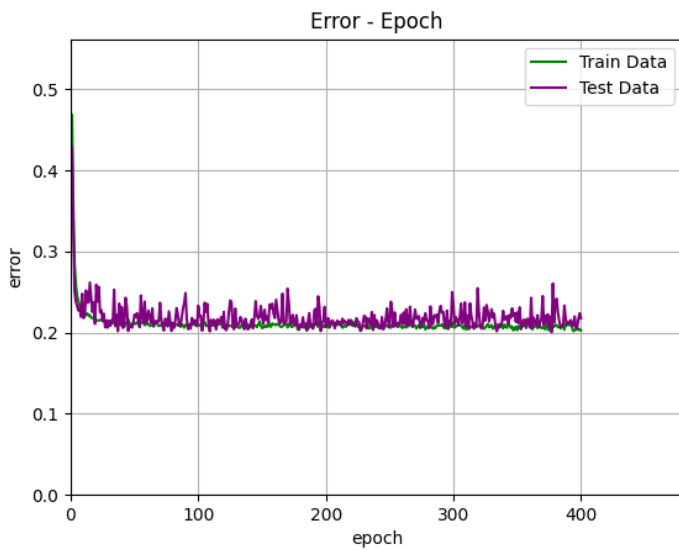
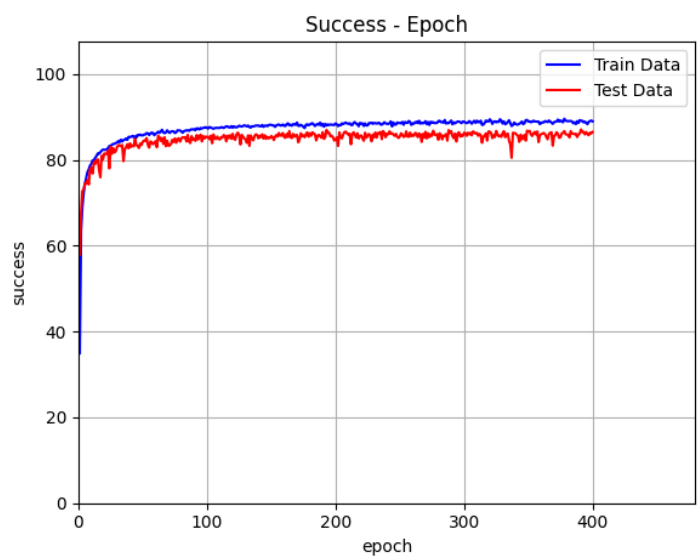
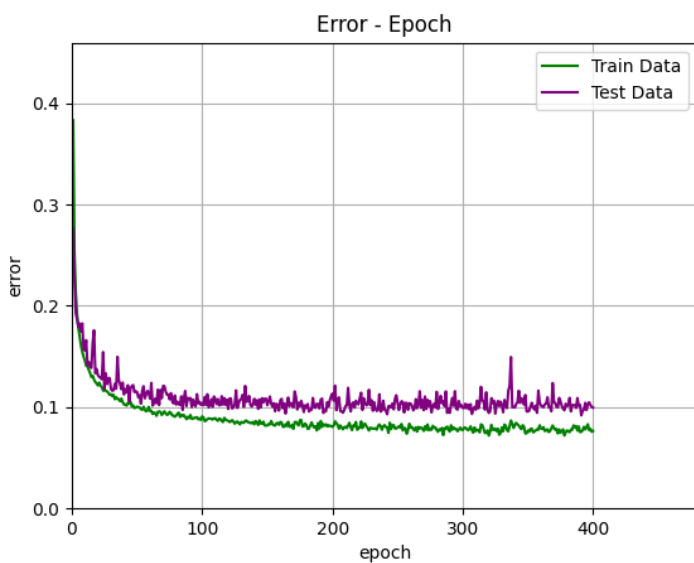


- **1st hidden layer = 10 2nd hidden layer = 10**



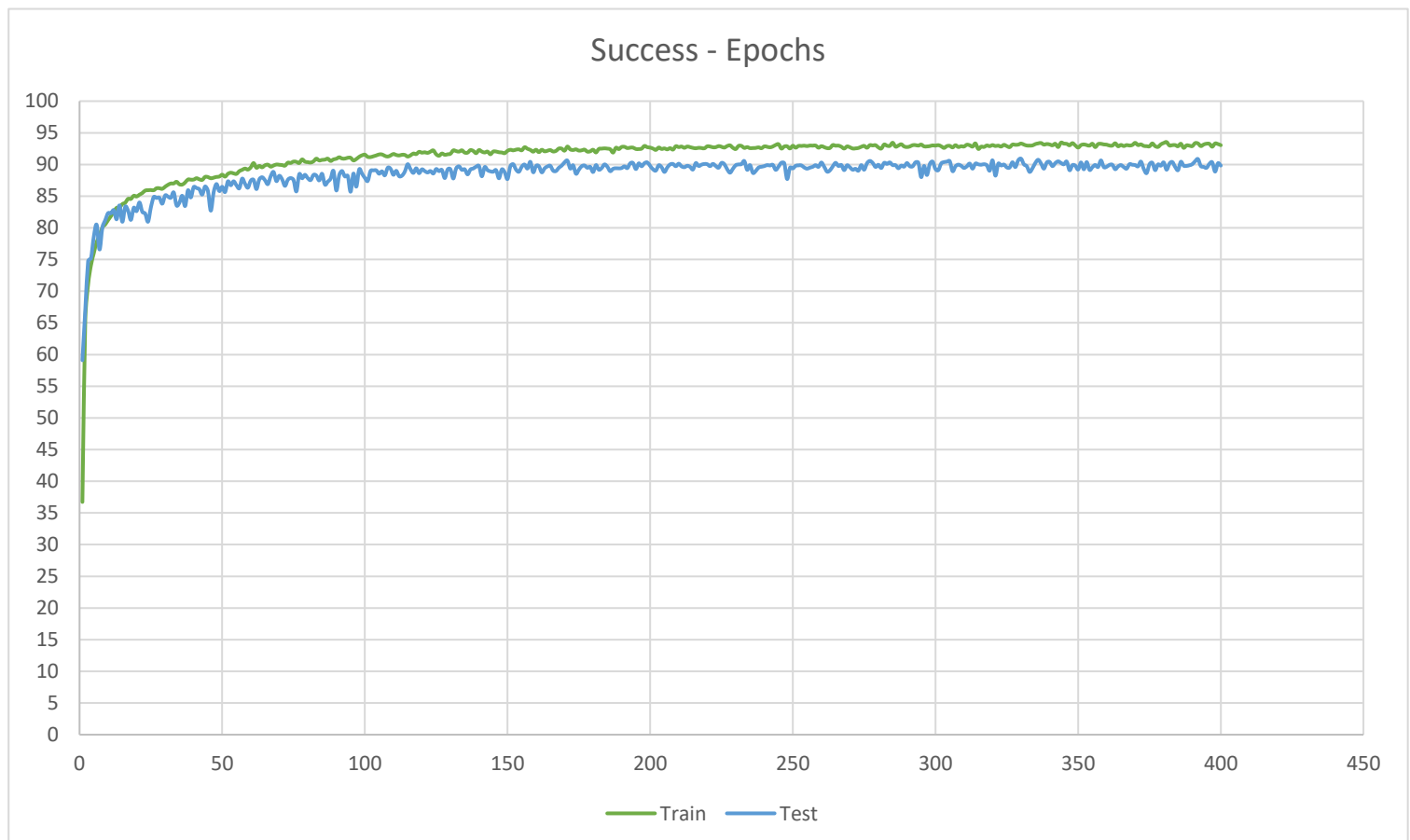
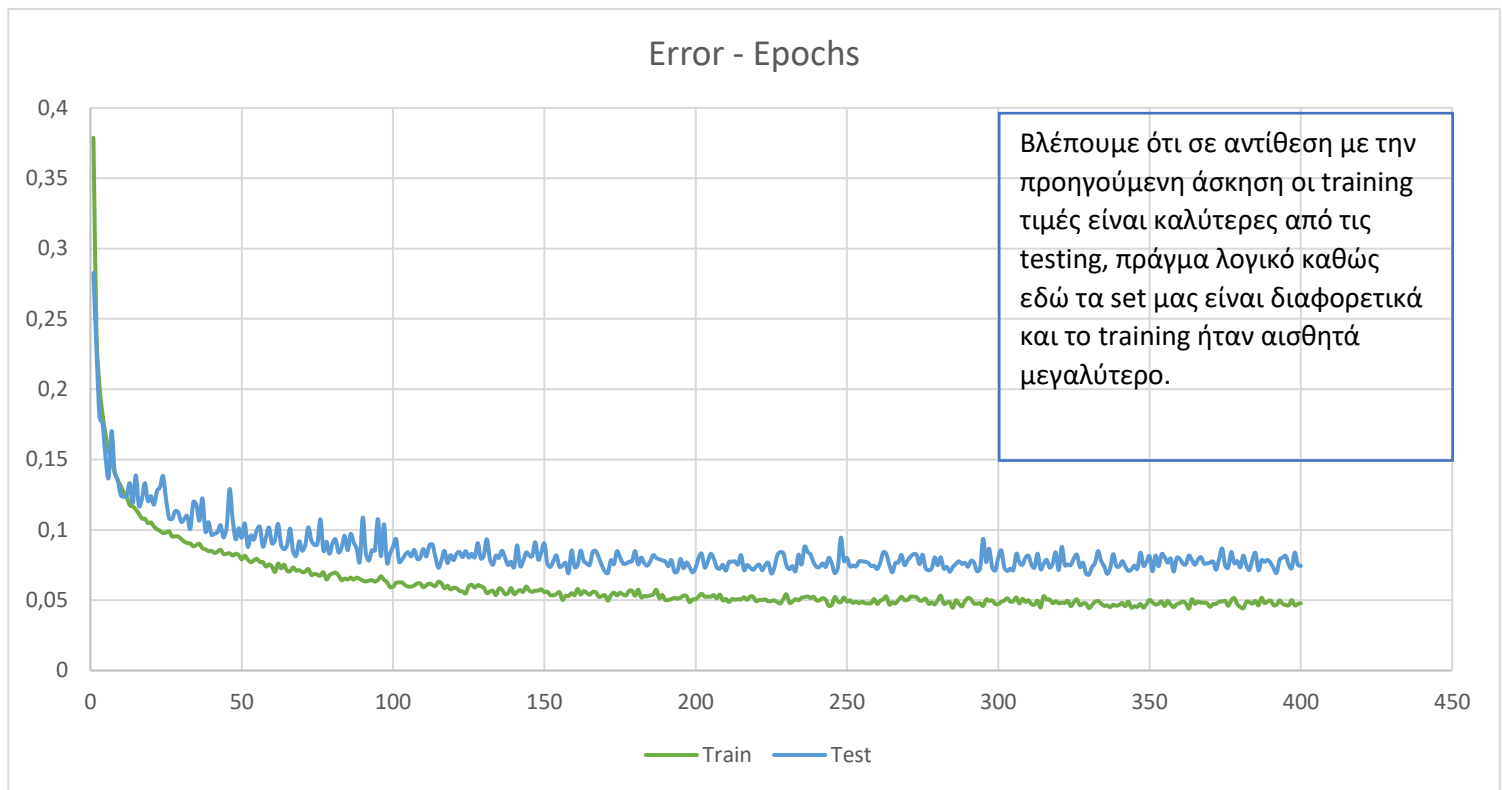
Παρατηρούμε ενώ έχουμε μια βελτίωση στις τιμές μας απέχουμε ακόμα αρκετά από το αποδεκτό φάσμα τιμών.

- **1st hidden layer = 50 2nd hidden layer = 30**



Με την παρούσα δομή βλέπουμε πως ξεκινάμε να έχουμε πιο αποδεκτά αποτελέσματα αλλά συνεχίζουμε να έχουμε πρόβλημα με το ποσοστό επιτυχίας.

- 1st hidden layer = 60 2nd hidden layer = 50



Εδώ βλέπουμε πως αυτή η δομή του νευρωνικού δικτύου καλύπτει πλήρως τις ανάγκες μας. Δοκίμασα να αυξήσω τους νευρώνες αλλά ο χρόνος εκτέλεσης αυξήθηκε κατακόρυφα (> 5 ώρες) και τα αποτελέσματα δεν ήταν καλύτερα από την τελευταία μας εκτέλεση. Επομένως καταλήγουμε ότι μια αρκετά καλή δομή δικτύου είναι η 16-60-50-26.

Επιλογή αριθμού εποχών

Όπως παρατηρήσαμε από όλες τις εκτελέσεις πιο πάνω τετρακόσιες εποχές είναι αρκετές, ίσως και παραπάνω από αρκετές. Ωστόσο, με τις τετρακόσιες εποχές φτάνουμε οριακά πάνω από το 94% κάτι που δεν συμβαίνει με λιγότερες εποχές.

Training and Testing sets

Η διαδικασία που ακολουθήθηκε για να δημιουργηθούν τα training και testing sets είναι η εξής (αφότου έγινε κανονικοποίηση ανά στήλη στο excel) :

1. Αλφαριθμητική ταξινόμηση των γραμμών
2. Σπάσιμο 70-30 κάθε γράμματος
3. 100 shuffles του κάθε αρχείου
4. Γράψιμο στα αντίστοιχα αρχεία

Τελικά αποτελέσματα εκτέλεσης

expected: O got: O
 expected: C got: C
 expected: Q got: F
 expected: A got: A
 expected: A got: A
 expected: L got: L
 expected: F got: F
 expected: P got: P
 expected: E got: E
 expected: Z got: Z
 expected: U got: U

Βλέπουμε ότι τα αποτελέσματα είναι αρκετά καλά (μέσα στα αποδεκτά όρια, 10/11 σωστά). Το πλήρες αρχείο βρίσκεται στο directory της άσκησης ως Results/trained_results.txt