Πρόβλημα 1. Με την μέθοδο Newton-Raphson μπορούμε να προσεγγίσουμε αριθμητικά την ρίζα μιας εξίσωση $f(x) = 0$ με την επαναληπτική αντικατάσταση

$$x \to x - \frac{f(x)}{f'(x)}$$

όπου $f'(x)$ είναι η πρώτη παράγωγος της $f(x)$. Η επαναληπτική διαδικασία συνεχίζεται μέχρι δυο διαδοχικές προσεγγίσης της ρίζας είναι αρκετά κοντά.

Ας υποθέσουμε ότι $f(x) = x^3 + x - 3$, δηλαδή θέλουμε να βρούμε τις ρίζες της $x^3 + x - 3 = 0$.

(α) Να γίνουν δυο ξεχωριστά function M-files με όνομα newtf και newtdf που να ορίζουν την $f(x)$ και την παράγωγό της, αντίστοιχα.

(β) Να γίνει ένα script με όνομα `newtongen.m` το οποίο :

1) Να ζητά στην οθόνη μια αρχική πρόβλεψη της ρίζας.
2) Να μετρά τον αριθμό των επαναλήψεων της μεθόδου με ένα μετρητή που να τον ονομάσετε steps.
3) Η επαναληπτική διαδικασία να σταματά όταν το απόλυτη σχετική διαφορά δυο διαδοχικών τιμών της προσέγγισης x, x' (δηλαδή η $|\frac{x-x'}{x}|$) είναι μικρότερη από 10^{-8} ή μετά από 20 steps.
4) Να τυπώνει στην οθόνη σε κάθε επανάληψη την προσεγγιστική τιμή x καθώς και την τιμή της συνάρτησης.
5) Όταν η επαναληπτική διαδικασία τερματίσει το script θα πρέπει αν η σχετική διαφορά των δυο διαδοχικών προσεγγίσεων είναι μικρότερη από 10^{-8}, να τυπώνει το μήνυμα `zero found at` και να τυπώνει το τελευταίο x, αλλιώς να τυπώνει `zero NOT found`.

Να τρέξετε το script με αρχικά δοσμένα $x = 1$, $x = i$, $x = -i$, και να βρείτε τις 3 ρίζες της εξίσωσης.

γ) Αλλάζοντας τις συναρτήσεις στα function M-files, να βρεθεί η ρίζα της εξίσωσης

$$\log x = \cos x$$

με αρχική προσέγγιση $x = 0.5$.

Πρόβλημα 2. Να υλοποιήσετε ένα script που να υπολογίζει τις ρίζες του τριωνύμου $ax^2 + bx + c = 0$ με την μέθοδο της διακρίνουσας. Η λογική δομή του script θα πρέπει να είναι η εξής

```
1.  Start
2.  Input data (a, b, c )
3.  If a = 0 then
     If b = 0 then
        If c = 0 then
         Display 'Solution indeterminate'
        else
         Display 'There is no solution'
        else
         x = -c/b
         Display x (only one root:  equation is linear)
      else if b^2<4ac then
        Display 'Complex roots'
      else if b^2=4ac then
        x= -b/(2a)
        Display x (equal roots)
      else
        x_1 = (-b + sqrt(b^2 - 4ac))/(2a)
        x_2 = (-b - sqrt(b^2 - 4ac))/(2a)
        Display x_1, x_2
4.  Stop.
```

Πρόβλημα 3. Να γίνει ένα script το οποίο να προσομοιώνει την ρίψη ενός νομίσματος 50 φορές. Το script θα πρέπει να τυπώνει στην οθόνη τι έφερε το νόμισμα σε κάθε ρίψη (Κ ή Γ) και στο τέλος να τυπώνει το πλήθος των κεφαλών και γραμμάτων στις 50 ρίψεις.