5η ΑΣΚΗΣΗ
ΑΡΙΘΜΗΤΙΚΗ ΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ ΙΙ

Στην άσκηση αυτή θα χρησιμοποιήσουμε την μέθοδο Runge-Kutta 4ης τάξης για να επιλύσουμε την εξίσωση του Newton για διάφορες περιπτώσεις δυνάμεων.
Για την περίπτωση δύο διαφορικών εξισώσεων της μορφής:

$$\frac{dx}{dt} = f_1(t,x,y), \quad \frac{dy}{dt} = f_2(t,x,y) \tag{1}$$

η μέθοδος δίνει:

$$\begin{aligned}
&K_{11} = f_1(t_i, x_i, y_i), && K_{21} = f_2(t_i, x_i, y_i) \\
&K_{12} = f_1(t_i + \frac{\tau}{2}, x_i + \frac{\tau}{2}K_{11}, y_i + \frac{\tau}{2}K_{21}), && K_{22} = f_2(t_i + \frac{\tau}{2}, x_i + \frac{\tau}{2}K_{11}, y_i + \frac{\tau}{2}K_{21}) \\
&K_{13} = f_1(t_i + \frac{\tau}{2}, x_i + \frac{\tau}{2}K_{12}, y_i + \frac{\tau}{2}K_{22}), && K_{23} = f_2(t_i + \frac{\tau}{2}, x_i + \frac{\tau}{2}K_{12}, y_i + \frac{\tau}{2}K_{22}) \\
&K_{14} = f_1(t_i + \tau, x_i + \tau K_{13}, y_i + \tau K_{23}), && K_{24} = f_2(t_i + \tau, x_i + \tau K_{13}, y_i + \tau K_{23}) \\
&t_{i+1} = t_i + \tau \\
&x_{i+1} = x_i + \frac{1}{6}\tau(K_{11} + 2K_{12} + 2K_{13} + K_{14}), && y_{i+1} = y_i + \frac{1}{6}\tau(K_{21} + 2K_{22} + 2K_{23} + K_{24})
\end{aligned} \tag{2}$$

Ο νόμος του Newton στη Μηχανική είναι διαφορική εξίσωση δευτέρου βαθμού ως προς το χρόνο μπορεί να γραφτεί ως σύστημα δύο διαφορικών πρώτου βαθμού:

$$\frac{dx}{dt} = v, \quad \frac{dv}{dt} = a(x,v)$$

Το σύστημα αυτό είναι ισοδύναμο με το (1) αν η μεταβλητή y είναι η ταχύτητα v, $f_2(t,x,y) = y$ και $f_2(t,x,y)$ είναι η επιτάχυνση.
Για επιτάχυνση θα πάρετε τις περιπτώσεις: α) μηδενική (ευθύγραμμη ομαλή κίνηση), β) σταθερή (ευθύγραμμη ομαλά επιταχυνόμενη κίνηση), γ) ανάλογη της $-v$ (επιβραδυνόμενη λόγω αντιστάσεων) και δ) ανάλογη του $-x$ (ταλάντωση). Καθε φορά επιλέξτε κατάλληλες αρχικές συνθήκες ($x(t=0)$ και $v(t=0)$). Για κάθε περίπτωση να σχεδιάσετε τα $x(t)$ και $v(t)$.
Προτάσεις για την σύνταξη του κώδικα.
Στο κυρίως πρόγραμμα θα ορίζετε τον τελικό και αρχικό (συνήθως 0) χρόνο, τον αριθμό βημάτων, αρχική θέση, την αρχική ταχύτητα και, αν θέλετε, επιλογή της επιτάχυνσης. Η subroutine Runge-Kutta θα καλείται με ορίσματα τον τελικό χρόνο, αρχική θέση και ταχύτητα και αριθμό βημάτων. Καλό θα είναι η πληροφορία για κάθε βήμα (χρόνος, θέση και ταχύτητα) να αποθηκεύεται σε αντίστοιχα array(που θα είναι και αυτά ορίσματα της subroutine) ώστε η εγγραφή σε αρχείο να γίνεται στο κυρίως πρόγραμμα (όπως προτάθηκε και στην προηγούμενη άσκηση).
Μπορείτε να έχετε μια ξεχωριστή subroutine που θα εκτελεί σε κάθε βήμα τις πράξεις που απαιτεί η μέθοδος. Για παράδειγμα:
SUBROUTINE RGCALC(T,X1,X2,DT)
όπου T, X1,X2 είναι ο χρόνος και η αρχική θέση και ταχύτητα κατά την είσοδο στην subroutine, ενώ στην έξοδο οι μεταβλητές αυτές θα έχουν τις νέες τιμές. DT είναι το χρονικό βήμα που θα ορίζεται στην subroutine Runge-Kutta απ΄ όπου και θα καλείται η subroutine RGCALC.