ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΟΜΕΑΣ ΘΕΩΡΗΤΙΚΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΜΑΘΗΜΑ : ΠΡΟΗΓΜΕΝΟΙ ΕΠΙΣΤΗΜΟΝΙΚΟΙ ΥΠΟΛΟΓΙΣΜΟΙ

1.12.2023

2η ΕΡΓΑΣΙΑ

Δίνονται οι αλγόριθμοι(σε μορφή ψευδοκώδικα) των μεθόδων

1. **Runge-Kutta-Fehlberg 4ης τάξης (RKF4)** και
2. **Adams-Predictor-Corrector 4ης τάξης (APC4)**

για την αριθμητική επίλυση ενός προβλήματος αρχικών τιμών (Π.Α.Τ) :

$$y'(t) = f(t, y(t)), \quad y(t_0) = y_0, \quad \text{όπου } y(t) \mid [a = t_0,\ b = t_F].$$

Α. Ζητείται να υλοποιηθούν σε γλώσσα προγραμματισμού C ή C++ (ή και σε MatLab) για την αριθμητική επίλυση των παρακάτω Π.Α.Τ. :

Εφαρμογές			
	Πρόβλημα Αρχικών Τιμών (Π.Α.Τ.)	**Διάστημα** $[a, b]$	**Αναλυτική λύση**
1.	$y' = 2t - y$, $\quad y(0) = -1$	$[0, 5]$	$y(t) = e^{-t} + 2t - 2)$
2.	$y' = 1 + (t - y)^2$, $\quad y(2) = 1$	$[2, 5]$	$y(t) = t + \dfrac{1}{1-t}$
3.	$y' = cos2t + sin3t$, $\quad y(0) = 1$	$[0, 10]$	$y(t) = \dfrac{1}{2}sin2t - \dfrac{1}{3}cos3t + \dfrac{4}{3}$
4.	$y' = (t + 2t^3)y^3 - ty$, $\quad y(0) = \dfrac{1}{3}$	$[0, 5]$	$y(t) = \dfrac{1}{\sqrt{3 + 2t^2 + 6e^{t^2}}}$
5.	$y' = 1 + \dfrac{y}{t} + (\dfrac{y}{t})^2$, $\quad y(1) = 0$	$[1, 4]$	$y(t) = t(tan(lnt))$

Να γίνει κατάλληλη πινακοποίηση των αποτελεσμάτων σας και να σχολιάσετε τα συμπεράσματά σας.

Β. (**Προαιρετικό**) Ζητείται να υλοποιηθεί ένας αλγόριθμος σε γλώσσα προγραμματισμού C ή C++ (ή και σε MatLab) για την αριθμητική επίλυση ενός Π.Α.Τ. ανώτερης τάξης ($m \geq 2$) (με τη κλήση ενός από τους ανωτέρω δοθέντες αλγορίθμους).

Εφαρμογές		
	Πρόβλημα Αρχικών Τιμών (Π.Α.Τ.) τάξης $m = 2$	**Ακριβής λύση**
1.	$y'' + y = 0$, $\quad y(0) = 2, \quad y'(0) = 3$	$y(t) = 2cost + 3sint$
2.	$t^2y'' - 2ty' + 2y = t^3lnt$, $\quad y(1) = 1, \quad y'(1) = 0$	$y(t) = \dfrac{7}{4}t + \dfrac{1}{2}t^3lnt - \dfrac{3}{4}t^3$

Να γίνει κατάλληλη πινακοποίηση των αποτελεσμάτων σας και να σχολιάσετε τα συμπεράσματά σας.

1. Αλγόριθμος **Runge-Kutta-Felhberg 4ης τάξης (RKF4)**

$/*$ Αριθμητική επίλυση ενός Π.Α.Τ. $*/$

$/*$ **Αρχικοποίηση** $*/$

begin

Διάβασε $t_0,\ t_F,\ y_0$, $/*$ Παράμετροι του Π.Α.Τ. $*/$

Rmax , $/*$ Παράμετρος ελέγχου ακρίβειας ($Rmax = 10^{-4}$) $*/$

ScaleMin, ScaleMax $/*$ Παράμετροι ελέγχου μεγέθους βήματος h $*/$

$/*$ ($ScaleMin = 0.1,\quad ScaleMax = 4.0$) $*/$

$h = (Rmax)^{\frac{1}{4}}$ $/*$ Αρχικό μέγεθος βήματος h $*/$

$Hmin = h * 10^{-4}$ $/*$ Ελάχιστο επιτρεπτό μέγεθος βήματος h $*/$

$t = t_0\ ;\ y = y_0$ $/*$ $(t,\ y)$: το τρέχον $(t_j,\ y_j)$ $*/$

$/*$ **Επανάληψη** $*/$

do

begin

if $t + h > t_F$ **then** $h = t_F - t$ $/*$ μέγεθος βήματος h για το τελικό βήμα$*/$

$/*$ **επόμενη εκτίμηση του σφάλματος** $\mathbf{e}[\mathbf{h}]$ $*/$

$m_1 = hf(t_j, y_j)$

$m_2 = hf(t_j + \frac{1}{4}h,\ y_j + \frac{1}{4}m_1)$

$m_3 = hf(t_j + \frac{3}{8}h,\ y_j + \frac{3}{32}m_1 + \frac{9}{32}m_2)$

$m_4 = hf(t_j + \frac{12}{13}h,\ y_j + \frac{1932}{2197}m_1 - \frac{7200}{2197}m_2 + \frac{7296}{2197}m_3)$

$m_5 = hf(t_j + h,\ y_j + \frac{439}{216}m_1 - 8m_2 + \frac{3680}{513}m_3 - \frac{845}{4104}m_4)$

$m_6 = hf(t_j + \frac{1}{2}h,\ y_j - \frac{8}{27}m_1 + 2m_2 - \frac{3544}{2565}m_3 + \frac{1859}{4104}m_4 - \frac{11}{40}m_5)$

$ErrorEstimate = \frac{1}{360}m_1 - \frac{128}{4275}m_2 - \frac{2197}{75240}m_4 + \frac{1}{50}m_5 + \frac{2}{55}m_6$ $/* \simeq$ $\mathbf{e}[\mathbf{h}]$ $*/$

$/*$ **Έλεγχος ακρίβειας** $*/$

$Ratio = |ErrorEstimate|/h$

if $Ratio < Rmax$ **then** $/*$ η ακρίβεια του επομένου y είναι αποδεκτή $*/$

begin

$t = t + h$ $/*$ $t = t_F$ για το τελικό βήμα $*/$

$y = y + \frac{25}{216}m_1 + \frac{1408}{2565}m_3 + \frac{2197}{4104}m_4 - \frac{1}{5}m_5)$ $/*$ $y \simeq y(t)$ $*/$

τύπωσε $(t,\ h,\ y)$

end

$/*$ **Προσδιορισμός του επομένου** $\mathbf{h}$ **:** $h * ScaleMin \leq next\ h \leq h * ScaleMax$ $*/$

$ScaleFactor = 0.84 * (Rmax/Ratio)^{1/4}$ **then**

if $ScaleFactor < ScaleMin$ **then** $ScaleFactor = ScaleMin$

if $ScaleFactor > ScaleMax$ **then** $ScaleFactor = ScaleMax$

$h = ScaleFactor * h$

end

while ($t \neq t_F$ or $h \geq Hmin$) $/*$ **κριτήριο διακοπής** $*/$

if $(t = t_F)$ **then τύπωσε** (t, y) $/*$ $y \simeq y(t_F)$ $*/$

else τύπωσε (συμβαίνει να είναι $h < Hmin$,

κάποια ιδιαιτερότητα παρατηρείται ”κοντά” στο τρέχον t .)

end.

2. Αλγόριθμος **Adams-Predictor-Corrector 4ης τάξης (APC4)**

$/*$ Αριθμητική επίλυση ενός Π.Α.Τ. $*/$

$/*$ **Αρχικοποίηση** $*/$

begin

Διάβασε $t_0,\ t_F,\ y_0$, $/*$ Παράμετροι του Π.Α.Τ. $*/$

$NumOfSteps$, $/*$ από το t_0 προς το t_f, j-βήμα : $t_j = t_0 + jh$ $*/$

$MaxIt$ $/*$ Μέγιστος αριθμός των επαναλήψεων του τύπου διόρθωσης $*/$

$NumSig$ $/*$ εκθέτης δύναμης του 10 $*/$

$h = (t_F - t_0)/NumOfSteps$ $/*$ το μέγεθος βήματος h $*/$

$RelTol = 10^{-NumSig}$ $/*$ η επιθυμητή ακρίβεια $*/$

$t = t_0\ ;\ y = y_0$ $/*$ $(t,\ y)$: το τρέχον $(t_j,\ y_j)$ $*/$

for $j = 0$ **to** 2 $/*$ αρχικές τιμές των $f_{-3},\ f_{-2},\ f_{-1}$ $*/$

begin

$f_{j-3} = f(t,\ y)$

$/*$ Χρήση μιας self-starting μεθόδου για τον υπολογισμό του y_{new} $*/$

$t = t + h;\ y = y_{new}$ $/*$ $y(t+h) \simeq y_{new}$ $*/$

τύπωσε $(j+1,\ t,\ y,\ f_{j-3})$

end

$/*$ Είναι $t = t_3,\ y = y_3$ και $f_{-3},\ f_{-2},\ f_{-1}$ είναι οι τιμές κλίσης στα $t_0,\ t_1,\ t_2$ αντίστοιχα. $*/$

$f = f(t, y)$ $/*$ $f = f_j = f(t_j, y_j)$ $*/$

$/*$ **Επανάληψη** $*/$

for $j = 3$ **to** (NumOfSteps-1)

begin

$p = y + \frac{h}{24}[-9f_{-3} + 37f_{-2} - 59f_{-1} + 55f]$ $/*$ **τύπος πρόβλεψης** $*/$

do

for $k = 1$ **to** $MaxIt$

begin

$c = y + \frac{h}{24}[f_{-2} - 5f_{-1} + 19f + 9f(t+h, p)]$ $/*$ **τύπος διόρθωσης** $*/$

$Delta = -\frac{19}{270}(c - p)$ $/*$ εκτίμηση του σφάλματος $\mathbf{e_c}[\mathbf{h}]$ $*/$

$p\ =\ c$ $/*$ προετοιμασία για την επόμενη επανάληψη $*/$

end

while $(|Delta| > RelTol * |c|)$ $/*$ **κριτήριο διακοπής** $*/$

if (το **κριτήριο διακοπής** δεν ικανοποιείται) **then**

τύπωσε (για την επιθυμητή ακρίβεια χρειάζεται μικρότερο μέγεθος βήματος h)

$/*$ **ενημέρωση** $*/$

$t = t + h;\ y = c$

$f_{-3} = f_{-2};\ f_{-2} = f_{-1};\ f_{-1} = f;\ f = f(t, y)$

τύπωσε $(j+1,\ t,\ y,\ f)$

end

if (το **κριτήριο διακοπής** ικανοποιείται στο κάθε βήμα) **then**

τύπωσε (οι τιμές y είναι ακριβείς σε $NumSig$ σημαντικά ψηφία)

end.

Οδηγίες για την υποβολή της 2ης Εργασίας

Σημείωση : Όλες οι υλοποιήσεις των ασκήσεων να γίνουν σε C (ή C++) (ή και σε MatLab).

Καταληκτική ημερομηνία υποβολής:

Η **2η ΕΡΓΑΣΙΑ** πρέπει εμπρόθεσμα να υποβληθεί ηλεκτρονικά στην e_class του μαθήματος μέχρι και την **Παρασκευή 29.12.2023** και **ώρα 23:59**.

Η **2η Εργασία** πρέπει να περιλαμβάνει:

A.
1. ένα αρχείο για το κάθε πρόγραμμα με όνομα **ergasia2_algorithm**, (όπου algorithm το όνομα του αλγορίθμου της μεθόδου (π.χ. ergasia2_RKF4), που θα περιέχει μόνο τον πηγαίο(source) κώδικα του αλγορίθμου της μεθόδου και
2. ένα μόνο αρχείο κειμένου(και για τις δύο μεθόδους) με όνομα **ergasia2_Απαντήσεις_xxxxxxx**(.tex σε latex ή .doc σε word) για την παρουσίαση των αποτελεσμάτων, των σχολίων και των συμπεράσμάτων σας.

B. Παρόμοια.

Για την υποβολή στην e_class πρέπει να επισυνάψετε ΜΟΝΟ ένα Φάκελο (συμπιεσμένο) με όνομα ERG_2_Ονοματεπώνυμο_xxxxx.zip ή (.rar), όπου xxxxxxx τα τελευταία ψηφία του Α.Μ. σας. Μέσα στον φάκελο αυτό να περιέχονται τα αρχεία με τον **πηγαίο**(source) **κώδικα** (και όχι εκτελέσιμα αρχεία) και το **αρχείο κειμένου** με την ανάλυση των αποτελεσμάτων.

Προσοχή: Είναι απαραίτητο στην αρχή του κάθε αρχείου (**κώδικα** και **κειμένου**) να αναγράφετε το ονοματεπώνυμό σας και τον ΑΜ.