Απαντήσεις στα προβλήματα, δεκτές ηλεκτρονικά μέσω ηλεκτρονικής αλληλογραφίας:

```
email: compPhysicsEKPA@gmail.com
subject: AMXXX...X
attachments: problemX.ipynb .OR. problemX.txt .OR. problemX.pdf
```

Ως θέμα τον αριθμό μητρώου (AMXXX...X) *του αποστολέα και επισυνημμένα είτε ένα αρχείο σε μορφή jupyter notebook* (∗.ipynb), *είτε ένα αρχείο κειμένου* (∗.txt) *που να περιέχει σχολιασμένο κώδικα σε οποιαδήποτε γλώσσα προγραμματισμού* **καθώς και το αποτέλεσμα της εκτέλεσής του (printout)**. *Το όνομα του αρχείου να έχει την μορφή problemX όπου X ο αριθμός του προβλήματος. Ο επισυναπτόμενος κώδικας να είναι εξολοκλήρου γραμμένος με λατινικούς χαρακτήρες (english/greeklish) –συμπεριλαμβανομένων και των σχολίων επί του ιδίου– προς αποφυγήν σφαλμάτων κωδικοποίησης.*

Πρόβλημα 1 (αποστολή μέχρι 07.10.19)

α) Χρησιμοποιώντας το υπόδειγμα της λογικής του κώδικα που παρατίθεται, να υπολογιστεί το $\epsilon > 0$ που ικανοποιεί την σχέση $1.0 + \epsilon = 1.0$ για αριθμούς κινητής υποδιαστολής διπλής ακρίβειας (double precision – binary64). Έπειτα, για $p(x) = x^2$, να διερευνηθεί η τιμή του λόγου

$$\left.\frac{p(x+n\epsilon)-p(x)}{n\epsilon}\right|_{x=2}, \tag{1}$$

για $n = 1, 2, 3, 4, 5, 50000, 50001, 50002, 50003$. Θεωρώντας ότι η εξ. (1) αποτελεί έναν αριθμητικό εκτιμητή της παραγώγου της $p(x)$ στο $x = 2$, να υπολογιστεί το σχετικό σφάλμα[1] για τις τιμές του n που δίνονται.

β) Να υπολογιστούν οι λόγοι:

(I) $(0.1 + 0.1 - 0.2)/\epsilon$

(II) $(0.1 + 0.2 - 0.3)/\epsilon$

(III) $(7./3. - 4./3. - 3./3.)/\epsilon$

Υπόδειγμα κώδικα για τον υπολογισμό του ϵ:

```
/*   C/C++   */
double e = 1.0;
while(1.0 + e != 1.0) e = 0.5*e;

# Python
e = 1.0
while(1.0 + e != 1.0): e = 0.5*e
```

[1] Γενικός ορισμός, αν $\hat{w}$ εκτιμητής του w, το απολυτό σχετικό σφάλμα του $\hat{w}$ είναι $|(w - \hat{w})/w|$.

Πρόβλημα 2 (αποστολή μέχρι 14.10.19)

Δίνεται η εξίσωση,

$$\tan x = \frac{x}{1-x^2}. \tag{2}$$

Να απαντηθεί τουλάχιστον ένα εκ των δύο ερωτημάτων:

α) Να λυθεί η εξ.(2) στο διάστημα $x \in [2, 4]$ χρησιμοποιώντας:

1) την μέθοδο της διχοτόμησης με 15 επαναλήψεις. *Θεωρήστε σαν τελική εκτίμηση της ρίζας (ρ) το μέσο του διαστήματος διχοτόμησης στην 15ή επανάληψη, δηλ.* $\hat{\rho} = 0.5(b_{14} + a_{14}) \approx \rho$, *με* $a_0 = 2$ *και* $b_0 = 4$.

2) την επαναληπτική σχέση[2]

$$x_{\mathrm{n}+1} = x_{\mathrm{n}} - f(x_{\mathrm{n}})/f'(x_{\mathrm{n}})$$

με

$$f(x) = \tan x - \frac{x}{1-x^2}$$

χρησιμοποιώντας $x_0 = 3.0$ και θεωρώντας σαν εκτιμητή της ρίζας το $x_6 = \hat{\rho} \approx \rho$.

Να εκτυπωθούν οι τιμές $\hat{\rho}$ και $f(\hat{\rho})$ για τις δυο περιπτώσεις. Ποια μέθοδος έδωσε $f(\hat{\rho})$ που να είναι περισσότερο συμβατό με το 0 ;

β) Να διερευνηθεί η ύπαρξη ριζών στο διάστημα $x \in [4, 10]$.
– *θέμα ελεύθερης ανάπτυξης ;-)*

Υπόδειγμα κώδικα:

```
/*   C/C++   */
#include "math.h"
double f(double x){return tan(x) - x/(1 - x*x);}
double df(double x){/* implement f'(x) */}
int main()
{
  double a = 2;
  double b = 4;
  double n = 0;
  double x = 3;
  while( n < 15 )
  {
    // ... implement  bisection logic
```

[2] Η σχέση αυτή είναι διάσημη με το όνομα Newton-Raphson.

```
        double c = 0.5*(a + b);

        // ... implement newton-raphson
        if ( n < 6 )
        {
           // x = x - f(x)/f'(x)
           // if (n == 5) ektypwsi twn x, f(x)
        }
        n = n + 1;
      }
    }

### /*   Python   */ ###
import numpy as np
import matplotlib.pyplot as plt # for exploratory graphics ;-)
def f(x): return np.tan(x) - x/(1-x**2)
def df(x): return 0. # implement here the derivative

# first: plot f(x) to get an idea how it varies
x = np.linspace(-10,10,100)  # an array with 100 steps for x [-10, 10]
y = f(x)
ax, fig = plt.subplots(figsize=(10,10))
plt.plot(x, y)               # plots f(x)
plt.plot(x, [0. for i in x]) # plots y = 0, i.e., x-axis
plt.show()

# logic for bisection/newtwon similar as for the C/C++ example
```

Πρόβλημα 3 (αποστολή μέχρι 19.12.19)

Να απαντηθεί τουλάχιστον ένα εκ των δύο ερωτημάτων:

α) Να υπολογιστούν οι τρεις πρώτες επαναλήψεις των μεθόδων Gauss-Seidel και Jacobi για το συστήμα $AX = b$, με

$$A = \begin{bmatrix} 2 & 2 & 3 \\ -1 & -3 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

και

$$b = \begin{bmatrix} 3 \\ 2 \\ 0 \end{bmatrix}$$

έχοντας σαν αρχική εκτίμηση το $X^{\mathrm{T}} = [0, 0, 0]^{\mathrm{T}}$.

β) Να λυθεί το σύστημα [3] $AX = b$, με

$$A = \begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix}$$

και

$$b = \begin{bmatrix} 4 \\ -1 \\ -5 \\ -2 \\ 2 \\ 2 \\ -1 \\ 1 \\ 6 \end{bmatrix}$$

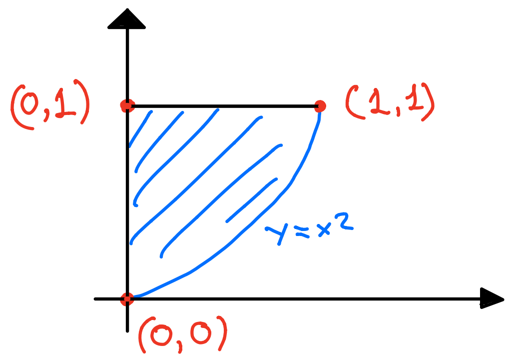
Να επαληθευεί η ορθότητα της λύσης (**απαραιτήτως πριν την αποστολή του αρχείου** στο compPhysicsEKPA@gmail.com) υπολογίζοντας τη συμβατότητα του $AX - b$ με το μηδενικό διάνυσμα (πίνακα-στήλη διαστάσεων 9×1).

[3] Πίνακες της μορφής A προκύπτουν κατά την διακριτοποίηση της εξίσωσης Poisson σε τετραγωνικό πλέγμα στις δυο διαστάσεις, βλέπε σημειώσεις μαθήματος – παράδειγμα με $\nabla^2\phi(x, y) = -\rho(x, y)/\epsilon$.

Πρόβλημα 4 (αποστολή μέχρι 09.01.20)

Να υπολογιστεί με μέθοδο Monte-Carlo (απλοϊκό ή απόρριψης) η μάζα των παρακάτω αντικειμένων και η αβεβαιότητά τους, για $N = 1000$ γεγονότα.

α) Δισδιάστατη πλάκα που οριοθετείται στην περιοχή $\{x \geq 0, y \leq 1, y \geq x^2\}$ (διαστάσεις μήκους σε μέτρα) με πυκνότητα $\rho(x, y) = \frac{20}{13}(x + y)$ [kg/m^3].



β) Κύβος πυκνότητας $\rho(x, y) = \frac{12}{31}(x^2 + yz)$ [kg/m^3] που οριοθετείται στην περιοχή $\{0 \leq x \leq 1, 1 \leq y \leq 2, 1 \leq z \leq 2\}$ με διαστάσεις μήκους μετρημένες σε μέτρα.

Δίνεται, προς σύγκριση, ο ακριβής υπολογισμός της μάζας των δυο σωμάτων είναι $M = 1\,$kg.