

Άσκηση 1

Υπολογιστική Φυσική I

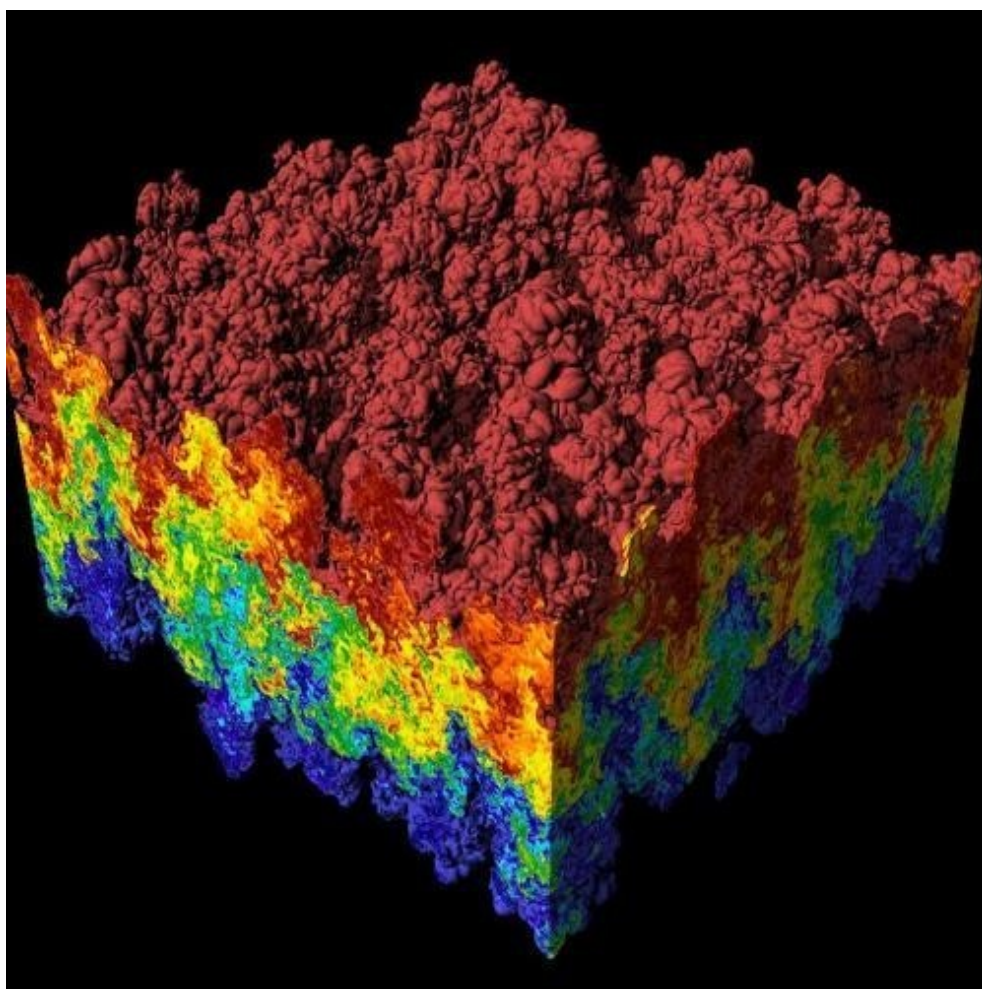
Σχολή: Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών

Ονοματεπώνυμο: Κοσμάς Φριτζάλας

Αριθμός Μητρώου Φοιτητή: ge18015

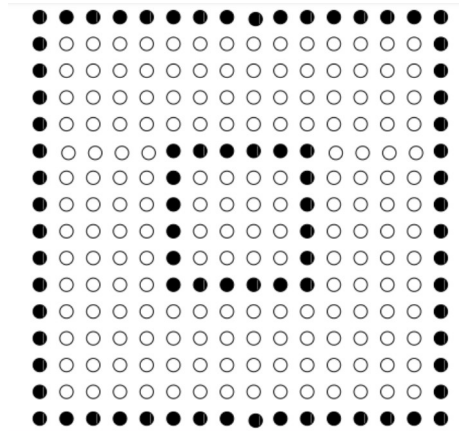
Όνομα Υπεύθυνου Άσκησης: Κωνσταντίνος Αναγνωστόπουλος

Ημερομηνία Διεξαγωγής Άσκησης: 2 Νοεμβρίου 2023



Ερώτημα 1

Σύμφωνα με τα ζητούμενα της άσκησης, πρέπει να μοντελοποιήσουμε το παρακάτω πρόβλημα ηλεκτροστατικής στον χώρο:



Εικόνα 1

Για την επίλυση του πρώτου ερωτήματος αλλά και για την μοντελοποίηση του συστήματος της εικόνας 1, αρχικά παίρνουμε τον ήδη έτοιμο κώδικα της διάλεξης 3 του μαθήματος της υπολογιστικής φυσικής και στην συνέχεια, αφού αλλάξουμε την ακρίβεια των δεκαδικών ψηφίων με το module της intrinsic και της έτοιμης συνάρτησης iso_fortran_env προσθέτουμε στην subroutine initialize_Lattice τις παρακάτω εντολές:

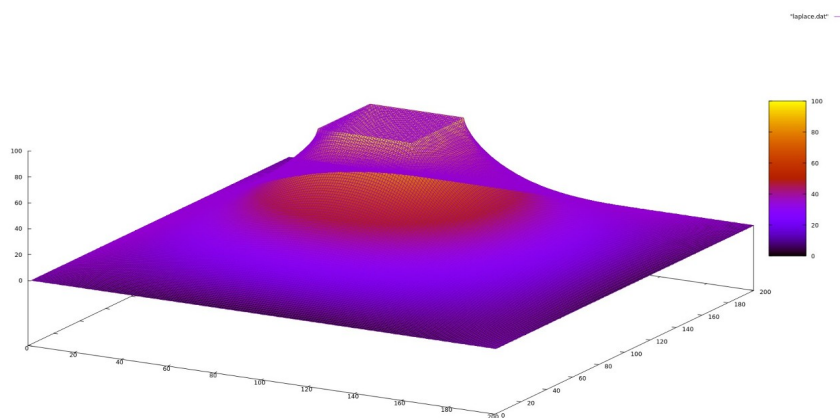
```
!=====
subroutine initialize_lattice(V,isConductor,L,V1,V2)
implicit none
!-----
!Declaration of variables:
integer                :: L
logical,dimension(L,L) :: isConductor
real(dp),dimension(L,L) :: V
real(dp)               :: V1,V2
integer                :: i,j  !For our loops two integers
!-----
!Calculations:
!We taking that every element in potencial V(x,y) has a zero initial value and also
that every place in lattice is not occupied:
V(:,:)=ZERO      !We taking that every element in the matrix has zero value. Also
because of contains internal procedure we don't need to declare the variable ZERO
again.
isConductor(:,:)=.FALSE. !In most places in our lattice there is not a conductor
!-----
!First we have the outer conductor:
```

```

do i=1,L
    V(1,i)=V1
    isConductor(1,i)=.TRUE.
    V(i,1)=V1
    isConductor(i,1)=.TRUE.
    V(i,L)=V1
    isConductor(i,L)=.TRUE.
    V(L,i)=V1
    isConductor(L,i)=.TRUE.
end do
!-----
!Next we have the inner conductor:
do i=(2*L/5),(3*L/5)
    V(2*L/5,i)=V2
    isConductor(2*L/5,i)=.TRUE.
    V(3*L/5,i)=V2
    isConductor(3*L/5,i)=.TRUE.
    V(i,2*L/5)=V2
    isConductor(i,2*L/5)=.TRUE.
    V(i,3*L/5)=V2
    isConductor(i,3*L/5)=.TRUE.
end do
!-----
end subroutine initialize_lattice
!=====

```

Με αυτόν τον τρόπο γίνεται η επιθυμητή μοντελοποίηση του προβλήματός μας. Αφού τρέξουμε τον κώδικα για $V_1=0V$ και $V_2=100V$ τότε θα πάρουμε την παρακάτω κατανομή για το δυναμικό του προβλήματος:



Εικόνα 2

Η παραπάνω εικόνα έχει αποθηκευτεί ως `test_image.png` και παριστάνει την κατανομή του δυναμικού για $L_1=200$ και μας δείχνει ότι έχει γίνει σωστή η μοντελοποίηση του προβλήματος μας μιας χ το δυναμικό στα άκρα του συστήματος παραμένει μηδενικό χ επίσης σχηματίζεται μια τετραγωνική περιοχή με σταθερό δυναμικό $V=100V$, το οποίο είναι σωστό με βάση την θεωρία μιας χ το δυναμικό στο εσωτερικό ενός αγωγού παραμένει σταθερό!

Στην συνέχεια για τον υπολογισμό των φορτίων Q_1 και Q_2 αλλά χ της χωρητικότητας C δημιουργούμε μια νέα subroutine την οποία ονομάζουμε `calc_charge_capacity` και την καλούμε στο κυρίως πρόγραμμα με την εντολή:

```
call calc_charge_capacity(V,L,sigma1,sigma2,Q1,Q2,V1,V2,C)
```

Επίσης αυτή η subroutine έχει την παρακάτω δομή:

```
subroutine calc_charge_capacity(V,L,sigma1,sigma2,Q1,Q2,V1,V2,C)

  implicit none
  !-----
  !Declaration of variables:
  integer          :: L
  real(dp),dimension(L,L) :: V,sigma1,sigma2
  real(dp)         :: C
  real(dp)         :: V1,V2
  real(dp)         :: Q1,Q2
  real(dp),parameter :: ONE=1.0_dp, ZERO=0.0_dp
  real(dp),parameter :: PI=atan2(ZERO,-ONE)
  real(dp),parameter :: PI_4=4*PI
  integer          :: i,j
  real(dp)         :: dummy
  !-----
  !Calculations:
  !We set initial values for our calculations:
  sigma1(:,:)=ZERO !We set all values of the surface density equal to zero because we
have zero charges to the free space
  sigma2(:,:)=ZERO
  !Calculate the surface density for the inner conductor and also the charge Q2 with the
help of a dummy index:
  !-----
  Q2=0.0_dp
  do i=(2*L/5),(3*L/5)
    dummy=-(V(2*L/5,i)-V((2*L/5)-1,i))/(PI_4)
```

```

    Q2=Q2+dummy
    dummy=-(V(3*L/5,i)-V((3*L/5)+1,i))/(PI_4)
    Q2=Q2+dummy
    dummy=-(V(i,2*L/5)-V(i,(2*L/5)-1))/(PI_4)
    Q2=Q2+dummy
    dummy=-(V(i,3*L/5)-V(i,(3*L/5)+1))/(PI_4)
    Q2=Q2+dummy
end do
!-----
!Calculate the surface density for the outter conductor and also the charge Q1 with the
help of a dummy index:
Q1=0.0_dp
do i=1,L
    dummy=-(V(1,i)-V(2,i))/(PI_4)
    Q1=Q1+dummy
    dummy=-(V(i,1)-V(i,2))/(PI_4)
    Q1=Q1+dummy
    dummy=-(V(i,L)-V(i,L-1))/(PI_4)
    Q1=Q1+dummy
    dummy=-(V(L,i)-V(L-1,i))/(PI_4)
    Q1=Q1+dummy
end do
!-----
!Calculation of the capacity:
C=ABS(Q1/(V1-V2))
end subroutine calc_charge_capacity

!=====

```

Επίσης γράφουμε στο κυρίως πρόγραμμα τις παρακάτω εντολές ώστε να γίνετε η εκτύπωση των φορτίων αλλά χ της χωρητικότητας:

```

print *,'#Q1=',Q1
print *,'#Q2=',Q2
print *,'#C=',C

```

Επίσης στην subroutine print_results τις προσθέτουμε τις καινούργιες μεταβλητές Q1, Q2 και C και στην συνέχεια αποθηκεύονται για λόγους χρήσιμους που θα δούμε σε επόμενο ερώτημα, σε ένα αρχείο με τίτλο charge_capacity.dat

Αφού τρέξουμε το πρόγραμμά μας για $L1=100$ και για μια τυχαία διαφορά δυναμικού ίση με 200, τότε έχουμε τα εξής αποτελέσματα:

```

kali@kali:~/Pylogistikiki_Fysiki_Uge18015-FRITZALAS-KOSMAS
File Actions Edit View Help
#Number of try: 6400 #Biggest error is: 1.1519986762214359E-008
#Number of try: 6407 #Biggest error is: 1.148973183262694E-008
#Number of try: 6408 #Biggest error is: 1.1459519555501174E-008
#Number of try: 6409 #Biggest error is: 1.1426426965214376E-008
#Number of try: 6410 #Biggest error is: 1.1398407548055378E-008
#Number of try: 6411 #Biggest error is: 1.1369451158316224E-008
#Number of try: 6412 #Biggest error is: 1.1339565738858154E-008
#Number of try: 6413 #Biggest error is: 1.1309779779367891E-008
#Number of try: 6414 #Biggest error is: 1.1280864882157683E-008
#Number of try: 6415 #Biggest error is: 1.1258421839221692E-008
#Number of try: 6416 #Biggest error is: 1.12284846258559357E-008
#Number of try: 6417 #Biggest error is: 1.1191374937888028E-008
#Number of try: 6418 #Biggest error is: 1.1161972676742725E-008
#Number of try: 6419 #Biggest error is: 1.1132641475342098E-008
#Number of try: 6420 #Biggest error is: 1.1101395536338212E-008
#Number of try: 6421 #Biggest error is: 1.1074206448753187E-008
#Number of try: 6422 #Biggest error is: 1.1045131832005884E-008
#Number of try: 6423 #Biggest error is: 1.101808484968011E-008
#Number of try: 6424 #Biggest error is: 1.0997158745767868E-008
#Number of try: 6425 #Biggest error is: 1.095827428798659E-008
#Number of try: 6426 #Biggest error is: 1.0929497387188285E-008
#Number of try: 6427 #Biggest error is: 1.0900162959595167E-008
#Number of try: 6428 #Biggest error is: 1.087128886703815E-008
#Number of try: 6429 #Biggest error is: 1.0843578479588174E-008
#Number of try: 6430 #Biggest error is: 1.0815871585921479E-008
#Number of try: 6431 #Biggest error is: 1.0786649795591075E-008
#Number of try: 6432 #Biggest error is: 1.075813351289602E-008
#Number of try: 6433 #Biggest error is: 1.07308047961268425E-008
#Number of try: 6434 #Biggest error is: 1.07018536525585464E-008
#Number of try: 6435 #Biggest error is: 1.067328343402479E-008
#Number of try: 6436 #Biggest error is: 1.0645678110816271E-008
#Number of try: 6437 #Biggest error is: 1.061771154736753E-008
#Number of try: 6438 #Biggest error is: 1.058981324593812E-008
#Number of try: 6439 #Biggest error is: 1.0561998393398446E-008
#Number of try: 6440 #Biggest error is: 1.0534236594139657E-008
#Number of try: 6441 #Biggest error is: 1.050653845154442E-008
#Number of try: 6442 #Biggest error is: 1.0478958369297521E-008
#Number of try: 6443 #Biggest error is: 1.0451481522884744E-008
#Number of try: 6444 #Biggest error is: 1.0423968361549689E-008
#Number of try: 6445 #Biggest error is: 1.0396561833658788E-008
#Number of try: 6446 #Biggest error is: 1.036926781758876E-008
#Number of try: 6447 #Biggest error is: 1.034199213552494E-008
#Number of try: 6448 #Biggest error is: 1.0314828997336938E-008
#Number of try: 6449 #Biggest error is: 1.0287735188249763E-008
#Number of try: 6450 #Biggest error is: 1.026078682591448E-008
#Number of try: 6451 #Biggest error is: 1.0233719649477288E-008
#Number of try: 6452 #Biggest error is: 1.0206846692214831E-008
#Number of try: 6453 #Biggest error is: 1.0180016829588531E-008
#Number of try: 6454 #Biggest error is: 1.0153272889934521E-008
#Number of try: 6455 #Biggest error is: 1.012642637488882E-008
#Number of try: 6456 #Biggest error is: 1.009988129572514E-008
#Number of try: 6457 #Biggest error is: 1.0073463848853947E-008
#Number of try: 6458 #Biggest error is: 1.0046968596818092E-008
#Number of try: 6459 #Biggest error is: 1.0020585258458688E-008
#Number of try: 6460 #Biggest error is: 9.994252536671418E-009
RQ1= -66.435382254666167
RQ2= -66.435310947779399
RQ3= -66.43531765162733884
kali@kali:~/Pylogistikiki_Fysiki_Uge18015-FRITZALAS-KOSMAS

```

Εικόνα 3

Τώρα θα τρέξουμε τον κώδικά μας για τις παρακάτω τιμές του δυναμικού στους αγωγούς:

- $V_1 = 0$, $V_2 = 200$
- $V_1 = -100$, $V_2 = 100$
- $V_1 = 200$, $V_2 = 400$

Έτσι για την περίπτωση α παίρνουμε τα αποτελέσματα της εικόνας 2. Για την περίπτωση β έχουμε τα εξής αποτελέσματα:

```

kali@kali:~/Pylogistikiki_Fysiki_Uge18015-FRITZALAS-KOSMAS
File Actions Edit View Help
#Number of try: 6208 #Biggest error is: 1.152456928781174E-008
#Number of try: 6209 #Biggest error is: 1.149625414399971E-008
#Number of try: 6210 #Biggest error is: 1.146806018075666E-008
#Number of try: 6211 #Biggest error is: 1.1438851789778001E-008
#Number of try: 6212 #Biggest error is: 1.140786879549921E-008
#Number of try: 6213 #Biggest error is: 1.137576877989774E-008
#Number of try: 6214 #Biggest error is: 1.13438932896964E-008
#Number of try: 6215 #Biggest error is: 1.131188922877631E-008
#Number of try: 6216 #Biggest error is: 1.12802293796964E-008
#Number of try: 6217 #Biggest error is: 1.125438812176584E-008
#Number of try: 6218 #Biggest error is: 1.122892338964658E-008
#Number of try: 6219 #Biggest error is: 1.120378923481954E-008
#Number of try: 6220 #Biggest error is: 1.118711213845458E-008
#Number of try: 6221 #Biggest error is: 1.1169586783897974E-008
#Number of try: 6222 #Biggest error is: 1.115029713986213E-008
#Number of try: 6223 #Biggest error is: 1.113084691188482E-008
#Number of try: 6224 #Biggest error is: 1.110984080859219E-008
#Number of try: 6225 #Biggest error is: 1.1088481877597071E-008
#Number of try: 6226 #Biggest error is: 1.106680848059219E-008
#Number of try: 6227 #Biggest error is: 1.0942827881226543E-008
#Number of try: 6228 #Biggest error is: 1.091386786223664E-008
#Number of try: 6229 #Biggest error is: 1.088489181223871E-008
#Number of try: 6230 #Biggest error is: 1.087762828280448E-008
#Number of try: 6231 #Biggest error is: 1.084998298102182E-008
#Number of try: 6232 #Biggest error is: 1.082845718938548E-008
#Number of try: 6233 #Biggest error is: 1.07928802923344E-008
#Number of try: 6234 #Biggest error is: 1.0763888884670374E-008
#Number of try: 6235 #Biggest error is: 1.072388872877692E-008
#Number of try: 6236 #Biggest error is: 1.07095479831228E-008
#Number of try: 6237 #Biggest error is: 1.068988129941101E-008
#Number of try: 6238 #Biggest error is: 1.066988129941101E-008
#Number of try: 6239 #Biggest error is: 1.06278476876006E-008
#Number of try: 6240 #Biggest error is: 1.0594838764872631E-008
#Number of try: 6241 #Biggest error is: 1.0560892745261344E-008
#Number of try: 6242 #Biggest error is: 1.053917113439518E-008
#Number of try: 6243 #Biggest error is: 1.051448848683584E-008
#Number of try: 6244 #Biggest error is: 1.048775915323881E-008
#Number of try: 6245 #Biggest error is: 1.046021897373358E-008
#Number of try: 6246 #Biggest error is: 1.04287796374221E-008
#Number of try: 6247 #Biggest error is: 1.0397923248974631E-008
#Number of try: 6248 #Biggest error is: 1.036483195654951E-008
#Number of try: 6249 #Biggest error is: 1.0330483288893878E-008
#Number of try: 6250 #Biggest error is: 1.029276878989923E-008
#Number of try: 6251 #Biggest error is: 1.025516662615121E-008
#Number of try: 6252 #Biggest error is: 1.02186686715564E-008
#Number of try: 6253 #Biggest error is: 1.021120939399801E-008
#Number of try: 6254 #Biggest error is: 1.01848807838171E-008
#Number of try: 6255 #Biggest error is: 1.015788611622658E-008
#Number of try: 6256 #Biggest error is: 1.013089947588788E-008
#Number of try: 6257 #Biggest error is: 1.0104837465729771E-008
#Number of try: 6258 #Biggest error is: 1.007705957805800E-008
#Number of try: 6259 #Biggest error is: 1.004881510221048E-008
#Number of try: 6260 #Biggest error is: 1.00211583751444E-008
#Number of try: 6261 #Biggest error is: 9.998437138171821E-009
RQ1= -66.43531377885277
RQ2= -66.43538807886127
RQ3= -66.4353275886382615
kali@kali:~/Pylogistikiki_Fysiki_Uge18015-FRITZALAS-KOSMAS

```

Εικόνα 4

Για την περίπτωση γ έχουμε τα εξής αποτελέσματα:

```

kali@kali: ~/Fpologistikhi_Fysiki_I/ge18015-FRITZALAS-KOSMAS
File Actions Edit View Help
#number of try: 6950 #Biggest error is: 1.152427218530824095E-000
#number of try: 6951 #Biggest error is: 1.14940808354168545E-000
#number of try: 6952 #Biggest error is: 1.1463839655334596E-000
#number of try: 6953 #Biggest error is: 1.143271662338267E-000
#number of try: 6954 #Biggest error is: 1.1403599318179861E-000
#number of try: 6955 #Biggest error is: 1.137368593021354E-000
#number of try: 6956 #Biggest error is: 1.1343786535700569E-000
#number of try: 6957 #Biggest error is: 1.1314000403217506E-000
#number of try: 6958 #Biggest error is: 1.1284214451734442E-000
#number of try: 6959 #Biggest error is: 1.1254524187809100E-000
#number of try: 6960 #Biggest error is: 1.1224983609281480E-000
#number of try: 6961 #Biggest error is: 1.1195425031473868E-000
#number of try: 6962 #Biggest error is: 1.1164093027241682E-000
#number of try: 6963 #Biggest error is: 1.1136640936371785E-000
#number of try: 6964 #Biggest error is: 1.1107431419077329E-000
#number of try: 6965 #Biggest error is: 1.1078213901782874E-000
#number of try: 6966 #Biggest error is: 1.1049110071326140E-000
#number of try: 6967 #Biggest error is: 1.1020086240869407E-000
#number of try: 6968 #Biggest error is: 1.099129744008116E-000
#number of try: 6969 #Biggest error is: 1.0962139640469104E-000
#number of try: 6970 #Biggest error is: 1.093137687052535E-000
#number of try: 6971 #Biggest error is: 1.0904045100581966E-000
#number of try: 6972 #Biggest error is: 1.0875965817476119E-000
#number of try: 6973 #Biggest error is: 1.0847429621207093E-000
#number of try: 6974 #Biggest error is: 1.0818437315121007E-000
#number of try: 6975 #Biggest error is: 1.0790415672090603E-000
#number of try: 6976 #Biggest error is: 1.0762050806979006E-000
#number of try: 6977 #Biggest error is: 1.0733742783486377E-000
#number of try: 6978 #Biggest error is: 1.0705548447731417E-000
#number of try: 6979 #Biggest error is: 1.067725411970456E-000
#number of try: 6980 #Biggest error is: 1.0649330306478078E-000
#number of try: 6981 #Biggest error is: 1.0621363344398568E-000
#number of try: 6982 #Biggest error is: 1.059329632219042E-000
#number of try: 6983 #Biggest error is: 1.0565543107077247E-000
#number of try: 6984 #Biggest error is: 1.053780318673172E-000
#number of try: 6985 #Biggest error is: 1.051003230269090E-000
#number of try: 6986 #Biggest error is: 1.0482404872121606E-000
#number of try: 6987 #Biggest error is: 1.045408087055253E-000
#number of try: 6988 #Biggest error is: 1.042740426643444E-000
#number of try: 6989 #Biggest error is: 1.039995827935713E-000
#number of try: 6990 #Biggest error is: 1.037267336882525E-000
#number of try: 6991 #Biggest error is: 1.034538854329237E-000
#number of try: 6992 #Biggest error is: 1.0318217391613871E-000
#number of try: 6993 #Biggest error is: 1.0291103080817265E-000
#number of try: 6994 #Biggest error is: 1.0264484521343921E-000
#number of try: 6995 #Biggest error is: 1.0237044989480637E-000
#number of try: 6996 #Biggest error is: 1.021010120940614E-000
#number of try: 6997 #Biggest error is: 1.018327115238313E-000
#number of try: 6998 #Biggest error is: 1.0156554708373731E-000
#number of try: 6999 #Biggest error is: 1.012963081509153E-000
#number of try: 7000 #Biggest error is: 1.0103235581402795E-000
#number of try: 7001 #Biggest error is: 1.0076632861455437E-000
#number of try: 7002 #Biggest error is: 1.005014322266200E-000
#number of try: 7003 #Biggest error is: 1.0023711638496025E-000
#number of try: 7004 #Biggest error is: 9.9973362921446096E-009
RQ1= -66.435303266783095
RQ2= -66.435310942401770
RC= 0.3321765163391954

kali@kali: ~/Fpologistikhi_Fysiki_I/ge18015-FRITZALAS-KOSMAS

```

Εικόνα 5

Παρατηρούμε ότι σε όλες τις περιπτώσεις που πήραμε ισχύει ότι $Q_1 = -Q_2$. Επίσης τα θετικά φορτία είναι συγκεντρωμένα στον αγωγό με το μικρότερο αριθμητικά δυναμικό, δηλαδή στις παραπάνω τρεις περιπτώσεις στον εξωτερικό τετράγωνο αγωγό. Αυτό οφείλεται κατά γνωστά στην ροή του ηλεκτρικού πεδίου από το υψηλότερο δυναμικό στο χαμηλότερο (δηλαδή τα θετικά φορτία ρέουν από το υψηλό δυναμικό στο χαμηλό) με αποτέλεσμα, γνωρίζοντας ότι το ηλεκτρικό πεδίο κινείται προς το χαμηλότερο δυναμικό, η συγκέντρωση του αρνητικού φορτίου θα είναι στην πλευρά με υψηλότερο δυναμικό. Επίσης ας πάρουμε ότι $V_1 = 0$ και $V_2 = 100$ ως πρώτη περίπτωση χ ως δεύτερη περίπτωση ότι $V_1 = 100$ και $V_2 = 200$. Για την πρώτη περίπτωση έχουμε τα εξής αποτελέσματα:

```

kali@kali: ~/Fpologistikhi_Fysiki_I/ge18015-FRITZALAS-KOSMAS
File Actions Edit View Help
#number of try: 6143 #Biggest error is: 1.128591146487899E-000
#number of try: 6144 #Biggest error is: 1.127536455980324E-000
#number of try: 6145 #Biggest error is: 1.1245232343653681E-000
#number of try: 6146 #Biggest error is: 1.121513207800314E-000
#number of try: 6147 #Biggest error is: 1.1185161324283080E-000
#number of try: 6148 #Biggest error is: 1.115525314555792E-000
#number of try: 6149 #Biggest error is: 1.1125418024081170E-000
#number of try: 6150 #Biggest error is: 1.1095404063003905E-000
#number of try: 6151 #Biggest error is: 1.1065906135095531E-000
#number of try: 6152 #Biggest error is: 1.103690295657700E-000
#number of try: 6153 #Biggest error is: 1.1007473874049553E-000
#number of try: 6154 #Biggest error is: 1.107743540431772E-000
#number of try: 6155 #Biggest error is: 1.1140808277089921E-000
#number of try: 6156 #Biggest error is: 1.1207830839039670E-000
#number of try: 6157 #Biggest error is: 1.1269941210277963E-000
#number of try: 6158 #Biggest error is: 1.1326407981700790E-000
#number of try: 6159 #Biggest error is: 1.138138280809294E-000
#number of try: 6160 #Biggest error is: 1.1430230991077210E-000
#number of try: 6161 #Biggest error is: 1.0973704113360500E-000
#number of try: 6162 #Biggest error is: 1.0940409091694040E-000
#number of try: 6163 #Biggest error is: 1.0915904624051941E-000
#number of try: 6164 #Biggest error is: 1.08923622505274E-000
#number of try: 6165 #Biggest error is: 1.0868027747272574E-000
#number of try: 6166 #Biggest error is: 1.0843800907810420E-000
#number of try: 6167 #Biggest error is: 1.081942267315161E-000
#number of try: 6168 #Biggest error is: 1.0795203178404504E-000
#number of try: 6169 #Biggest error is: 1.0770496931071187E-000
#number of try: 6170 #Biggest error is: 1.0745729400340304E-000
#number of try: 6171 #Biggest error is: 1.068804770262823E-000
#number of try: 6172 #Biggest error is: 1.064008040264226E-000
#number of try: 6173 #Biggest error is: 1.0632497974007921E-000
#number of try: 6174 #Biggest error is: 1.060450800970340E-000
#number of try: 6175 #Biggest error is: 1.057640374050340E-000
#number of try: 6176 #Biggest error is: 1.0548403080165510E-000
#number of try: 6177 #Biggest error is: 1.052126524030340E-000
#number of try: 6178 #Biggest error is: 1.049350917090031E-000
#number of try: 6179 #Biggest error is: 1.046500040000000E-000
#number of try: 6180 #Biggest error is: 1.043680080521541E-000
#number of try: 6181 #Biggest error is: 1.041187022135050E-000
#number of try: 6182 #Biggest error is: 1.0387152202200923E-000
#number of try: 6183 #Biggest error is: 1.036268000410107E-000
#number of try: 6184 #Biggest error is: 1.033840000000000E-000
#number of try: 6185 #Biggest error is: 1.031420000000000E-000
#number of try: 6186 #Biggest error is: 1.029000000000000E-000
#number of try: 6187 #Biggest error is: 1.026580000000000E-000
#number of try: 6188 #Biggest error is: 1.024160000000000E-000
#number of try: 6189 #Biggest error is: 1.021740000000000E-000
#number of try: 6190 #Biggest error is: 1.019320000000000E-000
#number of try: 6191 #Biggest error is: 1.016900000000000E-000
#number of try: 6192 #Biggest error is: 1.014480000000000E-000
#number of try: 6193 #Biggest error is: 1.012060000000000E-000
#number of try: 6194 #Biggest error is: 1.009640000000000E-000
#number of try: 6195 #Biggest error is: 1.007220000000000E-000
#number of try: 6196 #Biggest error is: 1.004800000000000E-000
#number of try: 6197 #Biggest error is: 9.981930041915404E-009
RQ1= 31.21564000320409
RQ2= 32.21760000110000
RC= 0.3321760000320007

kali@kali: ~/Fpologistikhi_Fysiki_I/ge18015-FRITZALAS-KOSMAS

```

Εικόνα 6

Υπολογιστική Φυσική Ι

Για την δεύτερη περίπτωση έχουμε τα εξής αποτελέσματα:

```

kali@kali: ~/Ppologistikiki_Fysiki_I/ge18015-FRITZALAS-KOSMAS
File Actions Edit View Help
Number of try: 6687 #Biggest error is: 1.1514822394188689E-008
Number of try: 6688 #Biggest error is: 1.1483734651935878E-008
Number of try: 6689 #Biggest error is: 1.1453579418230220E-008
Number of try: 6690 #Biggest error is: 1.1423488627943422E-008
Number of try: 6691 #Biggest error is: 1.1393439081075485E-008
Number of try: 6692 #Biggest error is: 1.1363511821845269E-008
Number of try: 6693 #Biggest error is: 1.1333611382724484E-008
Number of try: 6694 #Biggest error is: 1.1303825431241421E-008
Number of try: 6695 #Biggest error is: 1.1274153166596079E-008
Number of try: 6696 #Biggest error is: 1.1244488091950727E-008
Number of try: 6697 #Biggest error is: 1.1214922324143117E-008
Number of try: 6698 #Biggest error is: 1.1185428589754358E-008
Number of try: 6699 #Biggest error is: 1.1156040639273751E-008
Number of try: 6700 #Biggest error is: 1.1126729759788574E-008
Number of try: 6701 #Biggest error is: 1.1097483828776688E-008
Number of try: 6702 #Biggest error is: 1.1068294725391663E-008
Number of try: 6703 #Biggest error is: 1.1039198894734929E-008
Number of try: 6704 #Biggest error is: 1.1010288751115917E-008
Number of try: 6705 #Biggest error is: 1.0981267458915766E-008
Number of try: 6706 #Biggest error is: 1.0952362524258045E-008
Number of try: 6707 #Biggest error is: 1.0923599802481476E-008
Number of try: 6708 #Biggest error is: 1.0894939379597685E-008
Number of try: 6709 #Biggest error is: 1.0866244792858921E-008
Number of try: 6710 #Biggest error is: 1.0837689740733055E-008
Number of try: 6711 #Biggest error is: 1.0809202422024100E-008
Number of try: 6712 #Biggest error is: 1.0780788712593696E-008
Number of try: 6713 #Biggest error is: 1.0752444268291838E-008
Number of try: 6714 #Biggest error is: 1.0724164667408331E-008
Number of try: 6715 #Biggest error is: 1.0695998753362801E-008
Number of try: 6716 #Biggest error is: 1.0667865251826791E-008
Number of try: 6717 #Biggest error is: 1.0639837455528323E-008
Number of try: 6718 #Biggest error is: 1.0611878493488805E-008
Number of try: 6719 #Biggest error is: 1.0583968374788148E-008
Number of try: 6720 #Biggest error is: 1.0556163942965213E-008
Number of try: 6721 #Biggest error is: 1.0528424354561139E-008
Number of try: 6722 #Biggest error is: 1.0500741689575205E-008
Number of try: 6723 #Biggest error is: 1.0473087286380142E-008
Number of try: 6724 #Biggest error is: 1.0445683493288942E-008
Number of try: 6725 #Biggest error is: 1.0418132700261741E-008
Number of try: 6726 #Biggest error is: 1.0390749594080262E-008
Number of try: 6727 #Biggest error is: 1.0363404753027074E-008
Number of try: 6728 #Biggest error is: 1.0336172911073886E-008
Number of try: 6729 #Biggest error is: 1.03090837179467858E-008
Number of try: 6730 #Biggest error is: 1.02819222888671545E-008
Number of try: 6731 #Biggest error is: 1.0254922244712361E-008
Number of try: 6732 #Biggest error is: 1.0227921628753477E-008
Number of try: 6733 #Biggest error is: 1.0201063105241746E-008
Number of try: 6734 #Biggest error is: 1.0174233011639444E-008
Number of try: 6735 #Biggest error is: 1.0147516684774864E-008
Number of try: 6736 #Biggest error is: 1.0120857811239145E-008
Number of try: 6737 #Biggest error is: 1.0094197477883426E-008
Number of try: 6738 #Biggest error is: 1.0067788444694208E-008
Number of try: 6739 #Biggest error is: 1.00412219411585152E-008
Number of try: 6740 #Biggest error is: 1.0014844065153738E-008
Number of try: 6741 #Biggest error is: 9.9884971485117548E-009
Q1= -33.217649816828827
Q2= -33.217656687716789
C= 0.33217649816828828
kali@kali: ~/Ppologistikiki_Fysiki_I/ge18015-FRITZALAS-KOSMAS

```

Εικόνα 7

Παρατηρούμε πάλι ότι τα φορτία παρότι είναι διαφορετικά από τις προηγούμενες τρεις περιπτώσεις, δεν αλλάζουν αφού παραμένει σταθερή η διαφορά δυναμικού. Έτσι ανεξάρτητα από τα δυναμικά V_1 και V_2 , έχουμε ότι τα φορτία Q_1 και Q_2 αλλά κ η χωρητικότητα C του συστήματος, εξαρτώνται μόνο από την διαφορά δυναμικού V_1-V_2 .

BONUS Σχετικά με το ερώτημα 1 κ 2

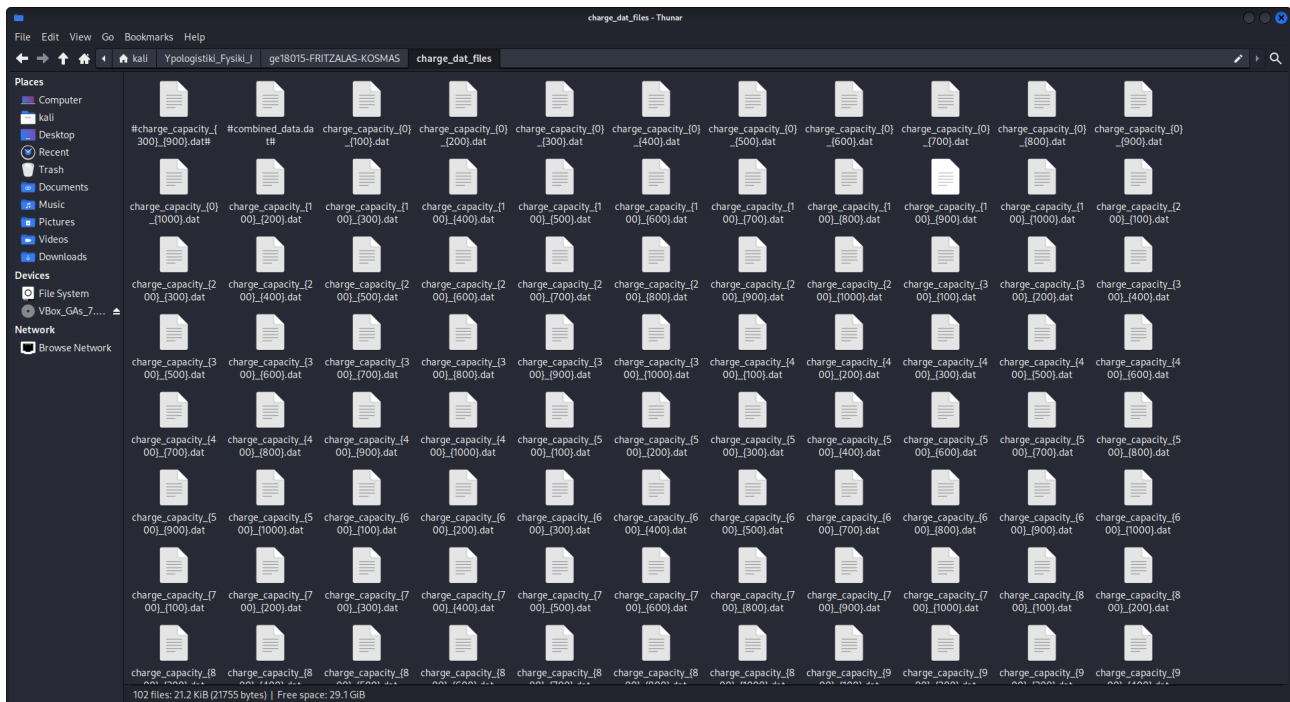
Παρότι “δείξαμε” εν μέρη ότι ισχύει $Q_1=-Q_2$ και ότι τόσο τα φορτία όσο κ η χωρητικότητα C του συστήματος εξαρτώνται μόνο από την διαφορά δυναμικού, καλό θα ήταν να δοκιμάζαμε για ένα εύρος τιμών της διαφοράς δυναμικού, ώστε να δούμε την παραπάνω συμπεριφορά. Έτσι δίνουμε την παρακάτω εντολή φλοιού, ώστε να τρέξει το κυρίως πρόγραμμα μας για διάφορες τιμές της διαφοράς δυναμικού:

```

# Loop for 'a-V1' from 0 to 1000 with a step of 100
for a in $(seq 0 100 1000); do
    # Loop for 'b' from 100 to 1000 with a step of 100
    for b in $(seq 100 100 1000); do
        if [ "$a" != "$b" ]; then
            (echo $a; echo $b; echo 1e-8; echo 100;)\./laplace; mv charge_capacity.dat
            charge_capacity_${a}_${b}.dat;
            # You can perform your desired operations here using 'a' and 'b'
        fi
    done
done

```


Έτσι το πρόγραμμά μας τρέχει για πολλές διαφορές δυναμικού χ κατασκευάζει τα αρχεία με όνομα $charge_capacity_V1_V2.dat$ με $V1=0,100,200,...,1000$ και $V2=100,200,300,...,1000$ (φυσικά έχουμε πάρει τον περιορισμό $V1=V2$). Έτσι φτιάχνονται τα παρακάτω αρχεία μας, τα οποία τα έχουμε σε ξεχωριστό φάκελο για να μην υπάρξει σύγχυση στον διδάσκοντα. Έτσι:



Εικόνα 8

Με βάση τις αλλαγές που έχουν γίνει στην subroutine `print_results`, τα αρχεία που τυπώνονται έχουν μια γραμμή χ 4 στήλες, όπου περιέχουν τα φορτία Q_1, Q_2 , την χωρητικότητα C και την διαφορά δυναμικού $ABS(V_1 - V_2)$. Ο σκοπός μας είναι να τα ενώσουμε σε ένα κοινό αρχείο χ μετά να γίνει το plot στο Gnuplot. Για αυτόν τον λόγο, εκμεταλλευόμενοι την εύκολη χρήση της βιβλιοθήκης Pandas της Python, μπορούμε να κάνουμε Inner Join τα αρχεία μας με το παρακάτω Script:

```
import os
```

```
import pandas as pd
```

```
# Specify the directory where your DAT files are located
```

```
directory = "/home/kali/Ypologistiki_Fysiki_I/ge18015-FRITZALAS-KOSMAS"
```

```
# Initialize an empty DataFrame to store the data
```

```
combined_data = pd.DataFrame()
```

Υπολογιστική Φυσική I

```

# Loop through all DAT files in the directory
for filename in os.listdir(directory):

    if filename.endswith(".dat"):

        file_path = os.path.join(directory, filename)

        # Read the DAT file into a DataFrame without changing columns
        data = pd.read_csv(file_path, sep='\t', header=None)

        # Append the data to the combined_data DataFrame
        combined_data = pd.concat([combined_data, data])

# Reset the index of the combined data
combined_data.reset_index(drop=True, inplace=True)

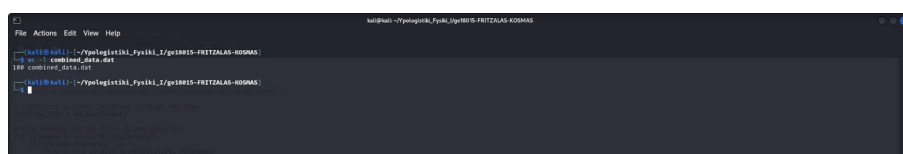
# Specify the path for the new DAT file
output_dat_file = "combined_data.dat"

# Save the combined data to a new DAT file
combined_data.to_csv(output_dat_file, sep='\t', header=False, index=False)

print(f"Combined data saved to {output_dat_file}")

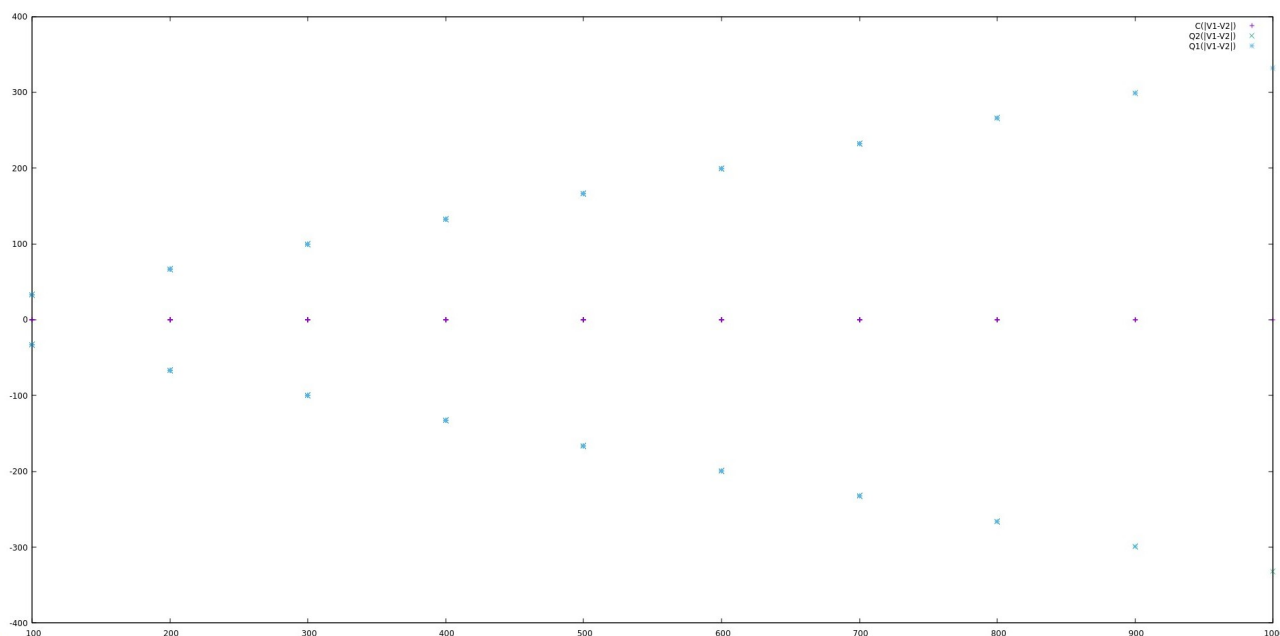
```

Έτσι σαν αποτέλεσμα μετά την εκτέλεση του στο φλοιό (python3 plot.py), να δημιουργηθεί ένα αρχείο με όνομα `combined_data.dat`. Με μια γρήγορη ματιά παρατηρούμε ότι έχει γίνει πετυχημένα το Inner Join και επίσης, όπως φαίνεται χ στην παρακάτω εικόνα, περιέχει σωστά 100 γραμμές (λόγω της διαμέρισης στα V_1 και V_2):



Εικόνα 9

Τώρα μπορούμε να κάνουμε σε κοινό διάγραμμα τα φορτία Q_1, Q_2 και την χωρητικότητα του συστήματος C συναρτήσει της διαφοράς δυναμικού $|V_1 - V_2|$. Έτσι θα έχουμε σχηματικά ότι:



Εικόνα 10

Παρατηρούμε όντως ανεξάρτητα από τα δυναμικά V_1 και V_2 , έχουμε ότι τα φορτία Q_1 και Q_2 αλλά η χωρητικότητα C του συστήματος, εξαρτώνται μόνο από την διαφορά δυναμικού $V_1 - V_2$. Βέβαια καταλήξαμε στην απάντηση του επόμενου ερωτήματος, διότι η σταθερή καμπύλη που περνάει από το σημείο 0.33 περίπου είναι η καμπύλη για την χωρητικότητα του συστήματος, γεγονός που δείχνει ότι η χωρητικότητα μάλιστα είναι ανεξάρτητη της διαφοράς δυναμικού. Τέλος εύκολα καταλήγουμε ότι ισχύει $Q_1 = -Q_2$.

Ερώτημα 2

Σε αυτό το ερώτημα θα τρέξουμε τον κώδικά μας για διαφορές δυναμικού $\delta V = |V_1 - V_2|$ ίσες με 100, 200, 500, 1000 ή 2000 και ταυτόχρονα θα κρατάμε τις τιμές των φορτίων Q_1 και Q_2 αλλά και της χωρητικότητας C του συστήματος. Για να το κάνουμε αυτό μπορούμε να δώσουμε την παρακάτω εντολή στον φλοιό:

```
for e in 100 200 500 1000 2000; do (echo 0; echo $e; echo 1e-6; echo 100) | ./laplace; mv charge_capacity.dat charge_capacity_${e}.dat; done
```

Έτσι θα τρέξει ο κώδικας για τις αντίστοιχες τιμές της διαφοράς δυναμικού και θα παράξει κάθε φορά για την εκάστοτε τιμή, το αντίστοιχο αρχείο με όνομα `charge_capacity_ δV .dat`. Στην

συνέχεια θέλουμε να ενώσουμε όλα τα παραπάνω αρχεία μας σε ένα. Η αντίστοιχη εντολή στον φλοιό είναι η παρακάτω:

```
for e in 100 200 500 1000 2000; cat charge_capacity_${e}.dat >> capacity_volt.dat
```

Αυτό έχει σαν αποτέλεσμα να φτιαχτεί ένα ενιαίο αρχείο με τις αντίστοιχες τιμές των φορτίων, της χωρητικότητας χ της διαφοράς δυναμικού. Έτσι έχουμε τις παρακάτω τιμές:

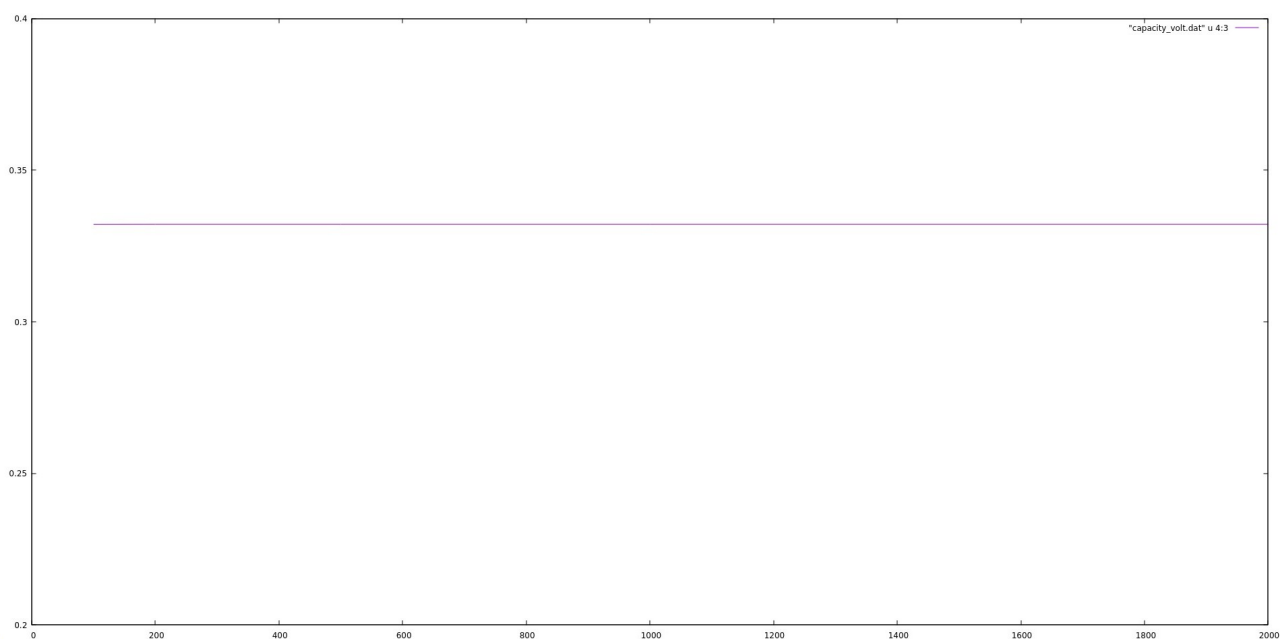
```

33.217124083711496      -33.217980553249322      0.33217124083711497      100.00000000000000
66.434779281888904      -66.433554248213904      0.33217389798544449      200.00000000000000
166.08774396759307      -166.08851027466910      0.33217548793518614      500.00000000000000
332.17601504371398      -332.17678764439449      0.33217601504371391      1000.00000000000000
664.35235876125598      -664.35332964075360      0.33217627938062799      2000.00000000000000
capacity_volt.dat (END)

```

Εικόνα 11

Παρατηρούμε ότι χ πάλι, όπως χ στο ερώτημα 1, υπάρχει εξάρτηση των φορτίων Q_1 και Q_2 από την διαφορά δυναμικού, αλλά χ ισχύει ότι $Q_1 = -Q_2$. Ωστόσο αυτό που παρατηρούμε επίσης είναι ότι η χωρητικότητα C του συστήματος είναι ανεξάρτητη από την διαφορά δυναμικού, αφού παραμένει σταθερή μέχρι το 50 δεκαδικό ψηφίο. Τώρα άμα κάνουμε την γραφική παράσταση $C(V = |V_1 - V_2|)$ θα έχει την παρακάτω μορφή:



Εικόνα 12

Ερώτημα 3

α) Τώρα μας δίνεται από την εκφώνηση ότι η χωρητικότητα C του συστήματος είναι συνάρτηση του L_1 που εκφράζει κατά τα γνωστά την διαμέριση του τετραγωνικού πλέγματος του προβλήματός μας. Θέλουμε να φτιάξουμε έναν κώδικα που να υπολογίζει τις τιμές της χωρητικότητας για μια τυχαία διαμέριση του αριθμού L_1 και στην συνέχεια να εκτυπώνει σε ένα αρχείο τύπου `.dat` τους αριθμούς L_1 με τους αντίστοιχους αριθμούς της χωρητικότητας C . Αυτό θα γίνει χρησιμοποιώντας τον ήδη υπάρχοντα κώδικα που έχουμε αλλά κάνοντας μια τροποποίηση στην subroutine `print_results`. Αυτό που θα κάνουμε είναι να προσθέσουμε ένα αρχείο με όνομα `C_L.dat` το οποίο θα αποθηκεύει απλά την τιμή της χωρητικότητας χ το μήκος L_1 . Έτσι θα έχουμε τον παρακάτω κώδικα:

!Have a file with the capacity of the system of conductors and the length of our system:

```
open(unit=13,file="C_L.dat")
write(13,*) L,C
close(13)
```

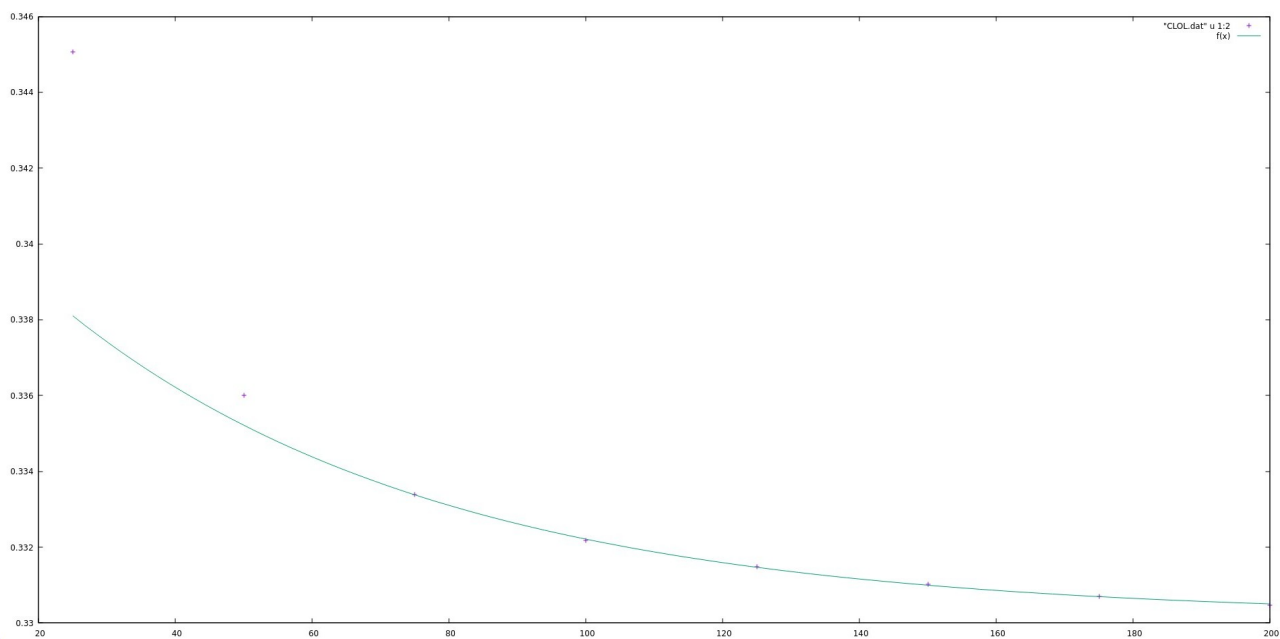
Στην συνέχεια θα τρέξουμε την διαμέριση για τα μήκη L_1 στον φλοιό με την παρακάτω εντολή:

```
for e in $(seq 25 25 200); do
(echo 0; echo 100; echo 1e-8;echo $e)|./laplace; mv C_L.dat C_L_$e.dat;
done
```


Με βάση τα παραπάνω αποτελέσματα, η συνάρτηση $C(L_1)$ συγκλίνει στην παρακάτω χωρητικότητα:

$$C = 0.33017$$

Επίσης η θεωρητική καμπύλη σε κοινό διάγραμμα με τα δεδομένα φαίνεται παρακάτω:



Εικόνα 16