



Εργαστήριο 3

Εαρινό Εξάμηνο 2012-2013

Στόχοι του εργαστηρίου

- Εντολές προσπέλασης της μνήμης
- Χρήση πινάκων

Χρήση μνήμης

Τα δεδομένα αποθηκεύονται σε ξεχωριστό τμήμα μνήμης το οποίο χρησιμοποιείται αποκλειστικά για αυτό το σκοπό. Γι' αυτό τον λόγο η δήλωση του πίνακα, που θα χρησιμοποιήσετε, πρέπει να γίνει στο τμήμα **.data**. Ο τρόπος δήλωσης του πίνακα και η σύνταξη των εντολών `load (lw)` και `store (sw)`, οι οποίες χρησιμοποιούνται για την μεταφορά δεδομένων μεταξύ μνήμη και καταχωρητή, εξηγούνται παρακάτω.

Δήλωση πίνακα

```
.data          # στο τμήμα .data πρέπει να δηλωθεί ο πίνακας  
  
Name_array: .space N      # Name_array: δώστε ότι όνομα  
                          # θέλετε στον πίνακα  
                          # .space X: δηλώνει ότι θέλουμε να  
                          # κρατήσουμε χώρο ίσο με N bytes για  
                          # τον πίνακα Name_array
```

Εντολή `load (lw)`

Η εντολή αυτή αποθηκεύει δεδομένα σε καταχωρητή, τα οποία έχει πάρει από συγκεκριμένη διεύθυνση της μνήμης.

```
Σύνταξη: lw Rt, Address (Rs)      Σημασία: Rt=Memory [Address+Rs]
```

Προσοχή: όπου Rt, Rs είναι καταχωρητές και όπου Address είναι το όνομα του πίνακα (label) που δώσατε στο τμήμα .data.

Με βάση την παραπάνω δήλωση (στο τμήμα .data) θα έπρεπε να γράψουμε:
`lw Rt, Name_array(Rs)`

Από την σύνταξη της εντολής αυτής γίνεται κατανοητό ότι ζητάμε τα δεδομένα της μνήμης στην διεύθυνση [Name_array+Rs] και τα οποία θα αποθηκευτούν στον καταχωρητή Rt.

Εντολή store (sw)

Η εντολή αυτή αποθηκεύει δεδομένα από έναν καταχωρητή σε συγκεκριμένη διεύθυνση στη μνήμη.

Σύνταξη: <code>sw Rt, Address(Rs)</code>	Σημασία: <code>Memory [Address+Rs]=Rt</code>
--	--

Ο τρόπος λειτουργίας είναι παρόμοιος με την εντολή **lw(load word)**. Η λειτουργία που υλοποιεί αυτή η εντολή είναι: Αποθήκευσε τα περιεχόμενα του καταχωρητή Rt στην μνήμη και συγκεκριμένα στην διεύθυνση **[Address+Rs]**.

Η διάφορα με την πρώτη εντολή είναι ότι η sw αποθηκεύει δεδομένα στην μνήμη ενώ η lw διαβάζει τα δεδομένα της μνήμης.

Ευθυγράμμιση

Όπως γνωρίζετε από την θεωρία ο MIPS είναι ένας 32bit επεξεργαστής. Αυτό συνεπάγεται ότι όλοι οι καταχωρητές του έχουν μέγεθος 32 bits ή αλλιώς 4 bytes. Ορίζουμε σαν **word** τα δεδομένα μεγέθους 4 bytes τα οποία χωράνε σε έναν register.

Όταν ορίζουμε έναν πίνακα στην μνήμη με την εντολή **.space** καθορίζουμε το μέγεθος του ίσο με έναν αριθμό από **bytes** όπως παρουσιάστηκε πριν. Εφόσον όμως τα στοιχεία που αποθηκεύονται σε αυτόν θα είναι μεγέθους 4 bytes το καθένα, πρέπει να καθορίσετε το μέγεθος του πίνακα να είναι πολλαπλάσιο του 4. Έτσι το πρώτο στοιχείο του πίνακα βρίσκεται στα bytes 0 έως 3. Αν θέλετε να το κάνετε load θα πρέπει να συντάξετε την εντολή `lw Rt, Address(Rs)` με τον Rs να έχει περιεχόμενο 0. Αν τώρα θέλετε το δεύτερο στοιχείο του πίνακα αυτό βρίσκεται στα bytes 4 έως 7. Για να το κάνετε load θα πρέπει να γράψετε την ίδια εντολή αλλά ο Rs θα πρέπει να έχει περιεχόμενο τον αριθμό 4.

Σημείωση: Όταν ορίζετε έναν πίνακα πρέπει να δηλώνετε και τι μέγεθος θα έχουν τα δεδομένα που θα αποθηκεύονται σε αυτόν. Αυτό γίνεται με την εντολή **.align n** όπου **n** έναν ακέραιος που θα δώσετε εσείς. Η εντολή αυτή σημαίνει ότι τα στοιχεία του πίνακα έχουν μέγεθος **2ⁿ**. Έτσι για έναν πίνακα 10 θέσεων με λέξεις 4 bytes πρέπει να συντάξετε την εντολή:

```
.align 2          #μέγεθος στοιχείου 4 bytes
label: .space 40  #μέγεθος πίνακα 40 bytes = 10 στοιχεία
```

Ένα παράδειγμα χρήσης μνήμης

```
.data
.align 2          # λέξεις 4 bytes
vector: .space 24 # πίνακας 6 θέσεων

.text
# ...

li $t1,4          # ένας τρόπος πρόσβασης στο 2° στοιχείο
lw $t0,vector($t1) # του πίνακα

# ...

la $t2,vector     # ακόμα ένας τρόπος για πρόσβαση στο 2°
lw $t3,4($t2)     # στοιχείο του πίνακα

# ...

la $t2,vector     # ακόμα ένας τρόπος για πρόσβαση στο 2°
addi $t2,$t2,4    # στοιχείο του πίνακα
lw $t0,0($t2)
```

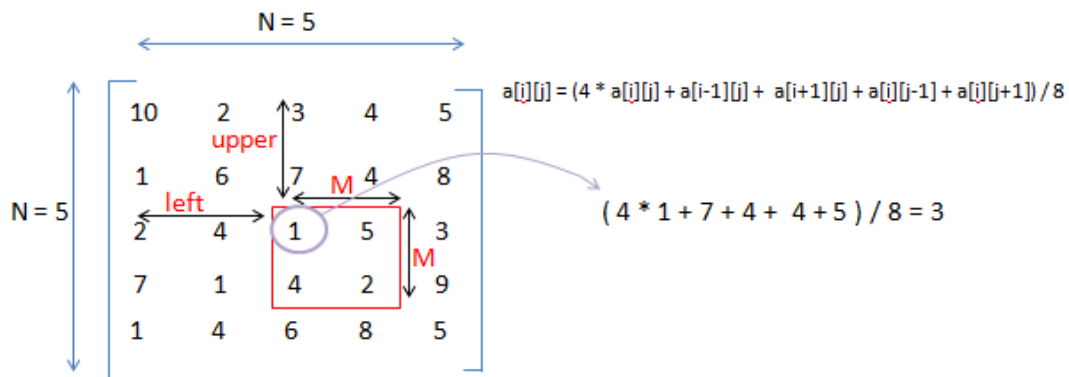
Για το επόμενο εργαστήριο έχετε να υλοποιήσετε την παρακάτω εργαστηριακή άσκηση. Την ώρα του εργαστηρίου θα εξετασθείτε προφορικά πάνω στον κώδικα που θα παραδώσετε.

Άσκηση 1 (10 μονάδες)

Να αναπτύξετε ένα πρόγραμμα που θα ζητάει αρχικά από τον χρήστη το μέγεθος (N) ενός τετραγωνικού πίνακα NxN και στη συνέχεια τα στοιχεία που θα αποθηκευτούν στις N² θέσεις αυτού του πίνακα. Ζητείται έπειτα το μέγεθος (M) ενός δεύτερου τετραγωνικού πίνακα MxM, ο οποίος είναι υποπίνακας εντός του αρχικού πίνακα. Ο χρήστης θα πρέπει επιπλέον να εισάγει την απόσταση του πρώτου στοιχείου του υποπίνακα από το αριστερό, αλλά και από το άνω άκρο του αρχικού πίνακα, ώστε να μπορεί να προσδιοριστεί η ακριβής θέση του υποπίνακα. Το πρόγραμμα θα αντιγράψει τον αρχικό πίνακα σε έναν νέο πίνακα, με τη διαφορά ότι στη θέση του υποπίνακα θα τοποθετεί έναν νέο υποπίνακα όπου κάθε στοιχείο του υπολογίζεται με την Εξίσωση 1. Πιο συγκεκριμένα, κάθε στοιχείο σύμφωνα με την Εξίσωση 1 αντικαθίσταται με το άθροισμα των τεσσάρων γειτονικών του στοιχείων (οριζόντια και κάθετα) συν το ίδιο το στοιχείο πολλαπλασιασμένο με 4 και όλο το άθροισμα διαιρεμένο με οκτώ (βλέπε παράδειγμα Εικόνα 1). Υποθέτουμε ότι ο εσωτερικός υποπίνακας δεν εφάπτεται στα όρια του μεγάλου πίνακα, ώστε να μην λάβουμε υπόψη τις περιπτώσεις που δεν υπάρχουν γείτονες σε κάποιες κατευθύνσεις. Τέλος, θα εκτυπώνεται ο τελικός πίνακας στην οθόνη. Η εμφάνιση πρέπει να γίνει έτσι, ώστε σε κάθε σειρά να εκτυπώνεται και μία γραμμή του πίνακα.

$$a[i][j] = (4 * a[i][j] + a[i-1][j] + a[i+1][j] + a[i][j-1] + a[i][j+1]) / 8$$

Εξίσωση 1



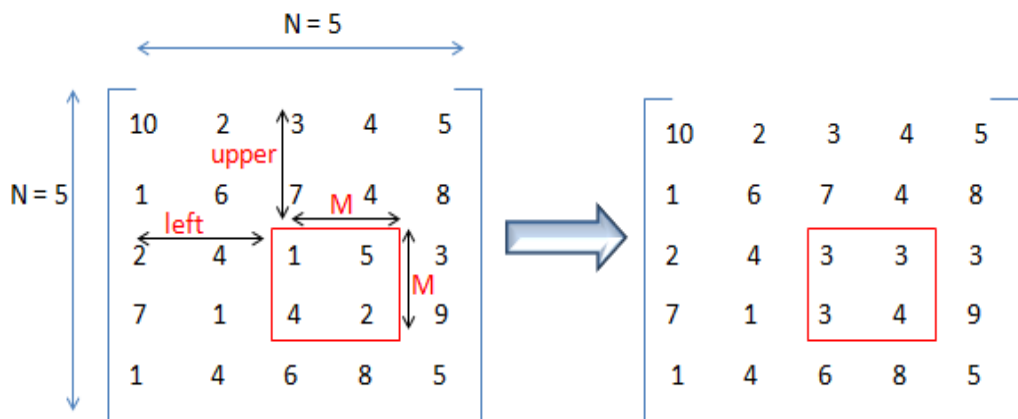
Εικόνα 1

Έστω ένας πίνακας 5x5 (N=5) και ένας υποπίνακας 2x2 (M=2) στο εσωτερικό του. Για να προσδιοριστεί η θέση του υποπίνακα σε σχέση με τον πίνακα θα πρέπει ο χρήστης να προσδιορίσει τις τιμές left και upper όπου εδώ είναι 2 και 2 αντίστοιχα.

Απεικονίζεται ο τύπος που χρησιμοποιείται για την αντικατάσταση του κάθε στοιχείου του υποπίνακα. Για παράδειγμα το στοιχείο 1 του υποπίνακα θα αντικατασταθεί από $(4 * 1 + 7 + 4 + 4 + 5) / 8$.

Σχηματικά:

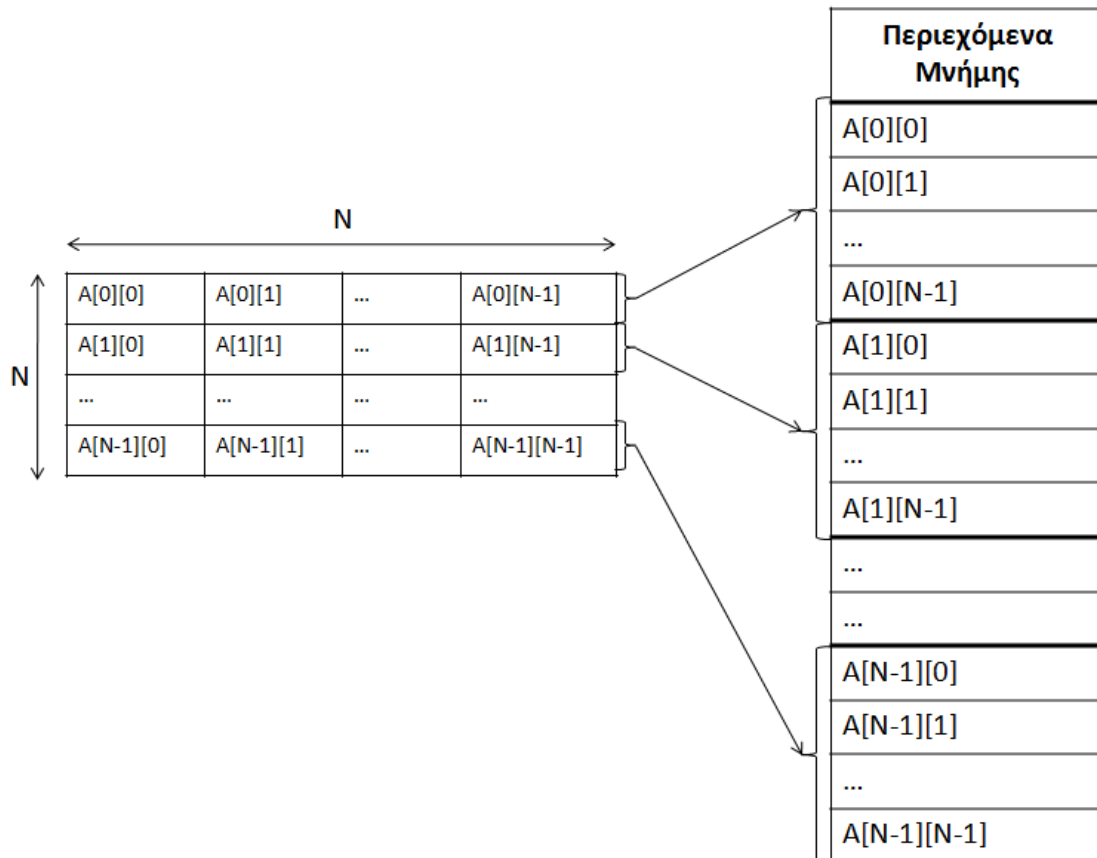
Έστω ένας πίνακας 5x5 ($N=5$) και ένας υποπίνακας 2x2 ($M=2$) στο εσωτερικό του. Για να προσδιοριστεί η θέση του υποπίνακα σε σχέση με τον πίνακα θα πρέπει ο χρήστης να προσδιορίσει τις τιμές *left* και *upper* όπου εδώ είναι 2 και 2 αντίστοιχα. Το πρόγραμμα θα πρέπει να δημιουργεί και να εκτυπώνει έναν καινούριο πίνακα ο οποίος είναι αντίγραφο του αρχικού, αλλά με τον υποπίνακα τα στοιχεία του οποίου έχουν υπολογιστεί με Εξίσωση 1 στην θέση του αρχικού υποπίνακα.



Υπόδειξη:

Έχουμε συζητήσει στην τάξη για το ότι μονοδιάστατοι πίνακες αποθηκεύονται στην μνήμη έτσι ώστε το στοιχείο $A[i]$ να βρίσκεται αμέσως μετά το στοιχείο $A[i-1]$ του πίνακα. Η αποθήκευση ενός πίνακα A δύο διαστάσεων γίνεται με το να αποθηκεύεται η γραμμή $i-1$ αμέσως πριν την γραμμή i .

Παράδειγμα πίνακα δύο διαστάσεων στην μνήμη ($N \times N$):



Ανάστροφος πίνακα

Ο ανάστροφος του πίνακα A είναι ο πίνακας A^T ο οποίος έχει γραμμές τις στήλες του A και στήλες τις γραμμές του A .

Δηλαδή, αν $A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$ τότε ο ανάστροφος είναι ο:

$$A^T = \begin{bmatrix} A_{11} & A_{21} & A_{31} \\ A_{12} & A_{22} & A_{32} \\ A_{13} & A_{23} & A_{33} \end{bmatrix}$$

Παράδειγμα εκτέλεσης:

```
Please define the dimensions of the matrix:
**** user input : 5
Now, you have to fill the matrix!
Please give an integer :
**** user input : 10
Please give an integer :
**** user input : 2
Please give an integer :
**** user input : 3
Please give an integer :
**** user input : 4
Please give an integer :
**** user input : 5
.
.
.
Please define the dimensions of the submatrix:
**** user input : 2
Please define the left distance:
**** user input : 2
Please define the upper distance:
**** user input : 2
The new matrix is:
10 2 3 4 5
1 6 7 4 8
2 4 3 3 3
7 1 3 4 9
1 4 6 8 5
```

Θα πρέπει να στέλνετε με email τις λύσεις των εργαστηριακών ασκήσεων σας στους διδάσκοντες στο ce134lab@gmail.com.

Το email σας θα πρέπει να περιέχει ως attachment **ένα zip file** με τον κώδικα σας.

Κάθε διαφορετική άσκηση στην εκφώνηση θα βρίσκεται και σε διαφορετικό asm file. **Το όνομα των asm files θα ΠΡΕΠΕΙ να αρχίζει με το ΑΕΜ σας.**

Για παράδειγμα, το lab2.zip θα περιέχει 3 asm files, ένα για κάθε μία από τις ασκήσεις του lab2, με ονόματα 999_lab2a.asm, 999_lab2b.asm, 999_lab2c.asm για τον φοιτητή με ΑΕΜ 999.

Το email σας θα έχει Subject: CE134, lab N, Section X (N ο αριθμός του lab, N=2 ..., και X=1 έως 7).

Το email σας θα έχει body: το όνομα σας και το ΑΕΜ σας.

Θα πρέπει να στέλνετε το email σας πριν βγείτε από την εξέταση του εργαστηρίου.