

Οικονομικό Πανεπιστήμιο Αθηνών  
Τμήμα Πληροφορικής  
Μάθημα: **Οργάνωση Συστημάτων Υπολογιστών**  
Χειμερινό Εξάμηνο 2023-24

Υπεύθυνη Εργασιών:  
Μαρία Τογαντζή  
Μέλος ΕΔΙΠ  
e-mail: [mst@aueb.gr](mailto:mst@aueb.gr)

Ημερομηνία: Δευτέρα, 18 Δεκεμβρίου 2023

## Εργασία 3<sup>η</sup> (Ομαδική)

Αραιοί ονομάζονται οι πίνακες (**Sparse Arrays**) που ένα μεγάλο ποσοστό των τιμών τους (συνήθως >80%) είναι μηδέν, για παράδειγμα:

### **Sparse Array (Μορφή A)**

| 0 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | -3 |

(μήκος πίνακα = 10) (μη μηδενικές τιμές = 2)

**A.** Ο παραπάνω πίνακας θα μπορούσε να δηλωθεί<sup>1</sup> ως:

```
int [] pin = new int [10];
```

Ένα υποπρόγραμμα που διαβάζει από την προκαθορισμένη είσοδο τις τιμές και δημιουργεί τον παραπάνω πίνακα, είναι το εξής:

```
static void readPin(int [] pin) {  
    for (int i=0; i<pin.length; i++) {  
        System.out.print ("Position " + i + " :");  
        pin [i] = in.nextInt();  
    }  
} //readPin
```

<sup>1</sup> Οι περιγραφές των υποπρογραμμάτων και των δηλώσεων είναι σε γλώσσα JAVA

Για παράδειγμα:

```
Reading Array A
Position 0 :0
Position 1 :0
Position 2 :0
Position 3 :0
Position 4 :0
Position 5 :8
Position 6 :0
Position 7 :0
Position 8 :0
Position 9 :-3
```

**B.** Το πρόβλημα είναι ότι οι πίνακες αυτοί χρησιμοποιούν μεγάλο χώρο μνήμης για την αποθήκευση μηδενικών τιμών. Ένας οικονομικός τρόπος αποθήκευσης τους είναι η αποθήκευση σε έναν μονοδιάστατο πίνακα μόνο των μη μηδενικών τιμών μαζί με τη θέση τους - στον αρχικό πίνακα:

### **Sparse Array (Μορφή B)**

| 5 | 8 | 9 | -3 |

(μήκος πίνακα = 4) (τιμές = 2)

Το παρακάτω υποπρόγραμμα δημιουργεί έναν αραιό πίνακα (μορφή B) με βάση τον αρχικό πίνακα (μορφή A):

```
static int createSparse (int [] pin, int [] Sparse) {
    int i, k=0;
    for (i=0; i<pin.length; i++)
        if (pin [i] != 0){
            Sparse [k++] = i;
            Sparse [k++] = pin [i];
        }
    return k;
} // createSparse
```

**Γ.** Η εμφάνιση στην προκαθορισμένη έξοδο των τιμών του αραιού πίνακα (μορφή B), γίνεται με το υποπρόγραμμα:

```
static void printSparse (int [] Sparse, int mikos) {
    for (int i=0; i<mikos;)
        System.out.println ("Position: "+ Sparse [i++] +
                             " Value: " + Sparse [i++]);
} //printSparse
```

Για παράδειγμα:

```
Displaying Sparse Array A
Position: 5 Value: 8
Position: 9 Value: -3
```

**Δ.** Έστω ότι θέλουμε το άθροισμα δύο αραιών πινάκων (μορφή B), του πίνακα A και του πίνακα B, δηλαδή να δημιουργήσουμε ένα νέο πίνακα C (μορφή B), του οποίου κάθε στοιχείο θα αποτελεί άθροισμα των αντίστοιχων στοιχείων των πινάκων A και B. Η δημιουργία ενός νέου πίνακα (SparseC) ως άθροισμα δύο αραιών πινάκων (SparseA και Sparse B), γίνεται με το υποπρόγραμμα:

```
static int addSparse (int [] SparseA, int mikosA, int [] SparseB, int mikosB, int []
SparseC)
{
    int a,b,c;
    for (a=0, b=0, c=0; a<mikosA && b<mikosB; )
        if (SparseA[a] < SparseB [b]) {
            SparseC [c++] = SparseA[a++];
            SparseC [c++] = SparseA[a++];
        }
        else if (SparseA[a] > SparseB [b]) {
            SparseC [c++] = SparseB[b++];
            SparseC [c++] = SparseB[b++];
        }
        else {
            SparseC [c++] = SparseA[a++];
            b++;
            SparseC [c++] = SparseA[a++] + SparseB[b++];
        }

    for (;a<mikosA;)
    {
        SparseC [c++] = SparseA[a++];
        SparseC [c++] = SparseA[a++];
    }
    for (;b<mikosB;)
    {
        SparseC [c++] = SparseB[b++];
        SparseC [c++] = SparseB[b++];
    }
    return c;
} //addSparse
```

Για παράδειγμα, αν οι δύο αραιοί πίνακες A και B είναι:

```
Displaying Sparse Array A
Position: 5 Value: 8
Position: 9 Value: -3
```

```
Displaying Sparse Array B
Position: 0 Value: 10
Position: 9 Value: 10
```

Η κλήση του υποπρογράμματος addSparse, θα είχε ως αποτέλεσμα τη δημιουργία ενός τρίτου αραιού πίνακα C ως άθροισμα – των αντίστοιχων στοιχείων - του πίνακα A και του πίνακα B.

```
Displaying Sparse Array C
Position: 0 Value: 10
Position: 5 Value: 8
Position: 9 Value: 7
```

Ε. Το παρακάτω υποπρόγραμμα εμφανίζει στο χρήστη μια λίστα επιλογών για να διαβάσει, να δημιουργήσει, να προσθέσει και να εμφανίσει αραιούς πίνακες:

```
static int readOption()
{
    System.out.println    ("\n-----");
    System.out.println    ("1. Read Array A");
    System.out.println    ("2. Read Array B");
    System.out.println    ("3. Create Sparse Array A" );
    System.out.println    ("4. Create Sparse Array B" );
    System.out.println    ("5. Create Sparse Array C = A + B");
    System.out.println    ("6. Display Sparse Array A");
    System.out.println    ("7. Display Sparse Array B");
    System.out.println    ("8. Display Sparse Array C");
    System.out.println    ("0. Exit");
    System.out.println    ("\n-----");
    System.out.print      ("Choice? ");
    int Answer=in.nextInt();
    return Answer;
} //readoption
```

**Ζητείται** να γράψετε στη συμβολική γλώσσα του επεξεργαστή **MIPS32**, τα παραπάνω υποπρογράμματα και ένα κύριο πρόγραμμα που καλεί τα υποπρογράμματα, ως εξής:

```
public static void main (String args[]) {

    int i, mikosA=0, mikosB=0, mikosC=0, op;
    int [] pinA = new int [10];
    int [] pinB = new int [10];
    int [] SparseA = new int [20];
    int [] SparseB = new int [20];
    int [] SparseC = new int [20];

    op=readOption();

    while((op>=1) && (op<=8)){
        switch (op) {
            case 1: System.out.println ("Reading Array A");
                    readPin(pinA);
                    break;
            case 2: System.out.println ("Reading Array B");
                    readPin(pinB);
                    break;
            case 3: System.out.println ("Creating Sparse Array A" );
                    mikosA= createSparse (pinA,SparseA);
                    System.out.println ( (mikosA/2) + " values" );
                    break;
            case 4: System.out.println ("Creating Sparse Array B" );
                    mikosB= createSparse (pinB,SparseB);
                    System.out.println ( (mikosB/2) + " values" );
                    break;
            case 5: System.out.println ("Creating Sparse Array C = A + B");
                    mikosC = addSparse (SparseA, mikosA, SparseB, mikosB, SparseC);
                    System.out.println ( (mikosC/2) + " values" );
                    break;
            case 6: System.out.println ("Displaying Sparse Array A");
                    printSparse(SparseA, mikosA);
                    break;
            case 7: System.out.println ("Displaying Sparse Array B");
                    printSparse(SparseB, mikosB);
                    break;
            case 8: System.out.println ("Displaying Sparse Array C");
                    printSparse(SparseC, mikosC);
                    break;
        }
        op=readOption();
    } //while
} //main
```

**Υλοποίηση:** Υλοποιείτε το κύριο πρόγραμμα και τα υποπρογράμματα όπως φαίνονται παραπάνω, στον κώδικα της java, με την ίδια λογική (αλγόριθμο), χρησιμοποιώντας για κάθε υποπρόγραμμα το ίδιο όνομα, τις ίδιες παραμέτρους και αποτελέσματα. Ειδικότερα, όπου στα υποπρογράμματα υπάρχει **παράμετρος κάποιος πίνακας** τότε να δίνεται ως παράμετρος η **διεύθυνση του πίνακα** (call by refernce).

Επίσης δηλώστε τους πίνακες με το ίδιο μέγεθος όπως και στη java (βλέπε main).

**Οδηγίες:**

- Η εργασία είναι ομαδική. Δικαίωμα υποβολής έχουν μόνο οι ομάδες που έχουν δημιουργηθεί για το σκοπό αυτό στο eclass. Υπενθυμίζεται ότι η προθεσμία για τη **δήλωση ομάδας** είναι μέχρι και την **Τετάρτη, 20/12/2023**.
- Τεκμηριώστε το πρόγραμμά σας με τα κατάλληλα **σχόλια**. Ως σχόλια, μπορείτε να χρησιμοποιείτε και εντολές από τον κώδικα java, όπως δίνεται παραπάνω.
- Το αρχείο που περιέχει το πλήρες πρόγραμμα (**class Sparse\_Array**) σε γλώσσα java, μπορείτε να το βρείτε στα έγγραφα – εργασίες του eclass.
- Στην πρώτη γραμμή του προγράμματος γράψτε, ως σχόλιο, **το όνομά σας, το επώνυμο και τον αριθμό του φοιτητικού σας μητρώου** (όλων των μελών της ομάδας).

Στα παραπάνω καλύτερα να αποφύγετε τη χρήση ελληνικών χαρακτήρων γιατί είναι δυνατό να δημιουργηθούν προβλήματα λόγω κωδικοποίησης των ελληνικών χαρακτήρων στο περιβάλλον του προσομοιωτή.

- Το πρόγραμμα πρέπει να εκτελείται με τη χρήση του προσομοιωτή **QtSpim**.
- Ονομάστε το αρχείο που περιέχει το πρόγραμμα σας με το όνομα **ergasia3.txt**
- Υποβάλλετε το αρχείο (ergasia3.txt), στο eclass του μαθήματος (Εργασίες - 3η Εργασία), μέχρι τη **Δευτέρα, 15/1/2024 (ώρα 23:59)**.
- Για τυχόν απορίες σας επικοινωνήστε με την κ. Μαρία Τογαντζή (mst@aueb.gr).