



Οικονομικό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής

Μάθημα: **Οργάνωση Συστημάτων Υπολογιστών**
Χειμερινό Εξάμηνο 2017-18

Διδάσκων: Αν. Καθηγητής Ι. Κουτσόπουλος

Υπεύθυνη Εργασιών:
Μαρία Τογαντζή
Μέλος ΕΔΙΠ
e-mail: mst@aueb.gr

Ημερομηνία δημοσίευσης: 2/11/2017
Προθεσμία υποβολής: 19/11/2017

1η Εργασία Προγραμματισμού σε Assembly

Η χρέωση σε ένα σταθμό στάθμευσης αυτοκινήτων είναι 5 ευρώ και 24 λεπτά. Υποθέστε ότι η χρέωση αυτή ισχύει για όλα τα οχήματα και ανεξάρτητα από τη διάρκεια της στάθμευσης.



Γράψτε ένα πρόγραμμα στη **συμβολική γλώσσα** του επεξεργαστή **MIPS32** που:

α. διαβάξει το ποσό (σε ευρώ και λεπτά) που πλήρωσε ο πελάτης, δηλαδή διαβάξει δύο ακεραίους αριθμούς, έναν ακέραιο που παριστά τα ευρώ και έναν ακέραιο που παριστά τα λεπτά του ευρώ. Υποθέστε ότι η μεγαλύτερη αξία σε ευρώ που δέχεται η εφαρμογή - άρα που μπορεί να δώσει ο πελάτης - είναι τα 20 ευρώ και η μεγαλύτερη αξία σε λεπτά του ευρώ είναι τα 99 λεπτά.

β. εμφανίζει τα ρέστα ως τον **ελάχιστο αριθμό χαρτονομισμάτων και κερμάτων** που αντιστοιχεί σ' αυτά. Τα διαθέσιμα χαρτονομίσματα για ρέστα είναι των 5 και 10 ευρώ και κέρματα των 1, 2, 5, 10, 20, 50 λεπτών, 1, 2 ευρώ.

A. Ακολουθούν περιπτώσεις εκτέλεσης της εφαρμογής:

Case 1 - Έλεγχος της εγκυρότητας των δεδομένων του προγράμματος:

```
-- Parking Ticket Payment --  
  
Fee: 5 euros and 24 cents  
  
Euros (<= 20): 50  
Cents (< 100): 0  
  
Error! Please try again.
```

Η μεγαλύτερη αξία σε ευρώ που δέχεται η εφαρμογή είναι τα 20 ευρώ και η μεγαλύτερη αξία λεπτών είναι τα 99 λεπτά!

Case 2 - Υπολογισμός και εμφάνιση των αποτελεσμάτων (ρέστα σε ευρώ και λεπτά)

```
-- Parking Ticket Payment --
```

```
Fee: 5 euros and 24 cents
```

```
Euros (<= 20): 20
```

```
Cents (<= 99): 0
```

```
Change:
```

```
1 x 10 euros
```

```
2 x 2 euros
```

```
1 x 50 cents
```

```
1 x 20 cents
```

```
1 x 5 cents
```

```
1 x 1 cents
```

Case 3 - Ακριβές αντίτιμο (ρέστα = 0)

```
-- Parking Ticket Payment --
```

```
Fee: 5 euros and 24 cents
```

```
Euros (<= 20): 5
```

```
Cents (< 100): 24
```

```
Change = 0
```

Case 4 - Δεν αρκούν τα χρήματα του πελάτη:

```
-- Parking Ticket Payment --
```

```
Fee: 5 euros and 24 cents
```

```
Euros (<= 20): 4
```

```
Cents (< 100): 0
```

```
Error! Not enough money!
```

B. Δίνεται η περιγραφή του αλγορίθμου που επιλύει το πρόβλημα:

Έλεγχξε την εγκυρότητα των δεδομένων του προγράμματος;

```
if (τα δεδομένα δεν είναι έγκυρα)
```

(case 1)

```
    εμφάνισε το μήνυμα "Error! Please try again.";
```

```
else {
```

```
    Υπολόγισε τα ρέστα σε λεπτά;
```

```
    if (ρέστα > 0)
```

(case 2)

```
        Για κάθε αξία νομίσματος από 20 ευρώ (2000 λεπτά) ως 1 λεπτό κάνε {
```

```
            Υπολόγισε και εμφάνισε την ποσότητά του νομίσματος στα ρέστα;
```

```
            Υπολόγισε την αξία του αμέσως μικρότερου νομίσματος;
```

```
        }
```

```
    else if (ρέστα == 0)
```

(case 3)

```
        εμφάνισε "Change = 0";
```

```
    else if (ρέστα < 0)
```

(case 4)

```
        εμφάνισε "Error! Not enough money!";
```

```
}
```

Γ. Οδηγίες Παράδοσης

- Η εργασία είναι ατομική.
- Ονομάστε το αρχείο που περιέχει το πρόγραμμα, με τον αριθμό του φοιτητικού σας μητρώου και κατάληξη .s (για παράδειγμα 3160000.s).
- Ακολουθείστε τις παρακάτω οδηγίες σχετικά με τον τρόπο συγγραφής του προγράμματος και την τεκμηρίωσή του:

α. Στην αρχή του προγράμματος πρέπει να υπάρχει σχόλιο με τις παρακάτω πληροφορίες:

```
# Author:
# Date:
# Description:
```

β. Μετά τη σύντομη περιγραφή αναφέρατε τους καταχωρητές που χρησιμοποιείτε και τη χρήση του καθενός.

γ. Πριν από κάθε ενότητα του προγράμματος περιγράψτε με τη μορφή ψευδοκώδικα το τμήμα του αλγορίθμου που υλοποιείτε.

```
# Pseudocode:
#
#
```

δ. Περιλάβετε επίσης σύντομα σχόλια σε επιμέρους εντολές ή ομάδες εντολών, για παράδειγμα:

```
printQuestion:
    li $v0, 4          # print(quest)
    la $a0, quest
    syscall
```

- Εκτελέστε το πρόγραμμά σας στον προσομοιωτή **SPIM** και βεβαιωθείτε ότι δεν έχει συντακτικά ή άλλα σφάλματα.
- **Συμπίεστε** (.zip ή .rar) το αρχείο που περιέχει το πρόγραμμα και **ανεβάστε** το στο eclass μέχρι την αναφερόμενη ημερομηνία.
- Όπως έχει ανακοινωθεί, η υλοποίηση της παρούσας εργασίας με επιτυχία, τεκμηριώνει δικαίωμα συμμετοχής στις άλλες δυο (2^η και 3^η εργασία).