



**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ**  
**ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ & ΥΛΙΚΟΥ**  
**ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ ΓΙΑ ΤΟ ΜΑΘΗΜΑ:**  
**ΗΡΥ 201 ΨΗΦΙΑΚΟΙ ΥΠΟΛΟΓΙΣΤΕΣ**

**ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2018-2019**

**Εργαστήριο 3:**

**Εμβάθυνση στη γλώσσα CLang – γνωριμία με την Assembly**

**1. Σκοπός του εργαστηρίου**

Σκοπός του εργαστηρίου είναι η εμβάθυνση στην κατανόηση λειτουργίας της μνήμης και στη γλώσσα CLang. Επίσης η γνωριμία με την Assembly.

**2. Περιγραφή προβλήματος**

Γράψτε ένα πρόγραμμα σε CLang (ξεκινήστε από C αν θέλετε, το πρόγραμμα όμως που πρέπει να φτιάξετε θα είναι σε CLang) το οποίο θα εκτελείται επαναληπτικά και κάθε φορά θα τρέχει μία από τις παρακάτω τέσσερις (4) επιλογές ανάλογα με την επιθυμία του χρήστη.

1) Διάβασμα ενός αριθμού N και εκτύπωση στην οθόνη της παρακάτω εξόδου:

```
1
1 2
1 2 3
1 2 3 4
.....
1 2 3 4 5 .... N
```

2) Διάβασμα ενός αριθμού N και εμφάνιση ενός μηνύματος που πληροφορεί αν ο αριθμός είναι μονός ή ζυγός.

3) Ανάγνωση πέντε ακεραίων από την κονσόλα και εκτύπωση των πενταπλάσιων τιμών τους.

4) Έξοδος

**3. Αναλυτικά η υλοποίηση του εργαστηρίου**

Υλοποιήστε τον κώδικα με συναρτήσεις. Η επεξεργασία για τις 3 πρώτες επιλογές να εκτελείται σε ξεχωριστή συνάρτηση. Αντίθετα, η εισαγωγή δεδομένων από τον χρήστη και η εκτύπωση μηνυμάτων να γίνεται στην κύρια συνάρτηση (main).

Παρακάτω ακολουθούν συγκεκριμένες οδηγίες για το τι πρέπει να εκτελείται σε κάθε συνάρτηση.

- **Κύρια συνάρτηση (main)**

Ο κώδικας στη main να τυπώνει στην οθόνη το μενού των δυνατοτήτων του χρήστη και να διαβάζει την επιλογή του. Στη συνέχεια να κάνει κλήση της κατάλληλης συνάρτησης περνώντας

τα κατάλληλα ορίσματα. Συγκεκριμένα:

- 1) Αρχικά, στην κονσόλα θα εμφανίζεται ένα μήνυμα το οποίο θα ενημερώνει τον χρήστη για τις επιλογές του.
- 2) Στη συνέχεια θα εμφανίζεται ένα μήνυμα όπου ο χρήστης θα πρέπει να εισαγάγει την επιλογή του.

**Please enter your choice (1-4):**

- 3) Ο χρήστης εισάγει την επιλογή του.
- 4) Στη συνέχεια θ' ακολουθεί η ανάγνωση των στοιχείων που απαιτούνται για την εκτέλεση καθενός από τα προγράμματα. Για την ανάγνωση των στοιχείων θα πρέπει να εμφανίζονται στον χρήστη τα αντίστοιχα μηνύματα ανάλογα με το πρόγραμμα που επέλεξε να τρέξει.
  - Συγκεκριμένα, για τις επιλογές 1 και 2 θα πρέπει το πρόγραμμα να ζητάει από το χρήστη τον αριθμό N.
  - Για την επιλογή 3 θα πρέπει να γίνει μία επαναληπτική διαδικασία όπου σε κάθε επανάληψη θα ζητάει και θα διαβάζει έναν ακέραιο αριθμό. Ο κάθε αριθμός θα αποθηκεύεται σε μία αλληλουχία θέσεων μνήμης.
- 5) Στη συνέχεια, θα γίνεται η κλήση της αντίστοιχης συνάρτησης.

**ΠΡΟΣΟΧΗ!!! Η κλήση των συναρτήσεων πρέπει να γίνεται με χρήση των κατάλληλων καταχωρητών ως ορίσματα. Πιο συγκεκριμένα, τα ορίσματα της κάθε συνάρτησης "περνάνε" στον κώδικα της συνάρτησης μέσω των καταχωρητών R4 έως και R7, ενώ το αποτέλεσμα της συνάρτησης επιστρέφει μέσω του καταχωρητή R2.**
- 6) Μετά την επιστροφή από τη συνάρτηση, θα εκτυπώνεται το κατάλληλο μήνυμα από το πρόγραμμα που εκτελέστηκε.
- 7) Θα γίνεται επιστροφή στο αρχικό μενού επιλογών.

#### • Συνάρτηση #1

- 1) Η συνάρτηση αυτή θα δέχεται ως όρισμα την τιμή N.
- 2) Μέσω κατάλληλων βρόγχων επανάληψης θα γίνεται η εκτύπωση στην οθόνη της παρακάτω δομής:

```
1
1 2
1 2 3
1 2 3 4
.....
1 2 3 4 5 .... N
```

- 3) Θα γίνεται επιστροφή από τη συνάρτηση του συνολικού αριθμού των στοιχείων που εκτυπώθηκαν.
- 4) Εκτύπωση (στη main) της επιστρεφόμενης τιμής.

#### • Συνάρτηση #2

- 1) Η συνάρτηση αυτή θα δέχεται ως όρισμα την τιμή N.
- 2) Μέσω κατάλληλων βρόγχων επανάληψης θα γίνεται η αναγνώριση του αριθμού αν είναι μονός ή ζυγός. **Μην χρησιμοποιήσετε την εντολή % (modulus).**
- 3) Θα γίνεται η επιστροφή μίας κατάλληλης τιμής στη main που να καθορίζει αν ο αριθμός είναι μονός ή ζυγός.
- 4) Ανάλογα με την επιστρεφόμενη τιμή θα γίνεται η εκτύπωση ενός κατάλληλου μηνύματος στη main σχετικά με τη "φύση" του αριθμού.

#### • Συνάρτηση #3

- 1) Η συνάρτηση αυτή θα δέχεται ως ορίσματα **2 διευθύνσεις θέσεων μνήμης**. Η πρώτη διεύθυνση είναι ο χώρος μνήμης που είναι αποθηκευμένες οι τιμές προς επεξεργασία. **Οι τιμές θα διαβάζονται από το χρήστη με μια επαναληπτική διαδικασία και θα αποθηκεύονται σε αυτό το χώρο στη συνάρτηση main.** Η δεύτερη διεύθυνση είναι ο χώρος μνήμης που θα αποθηκευθούν τα τελικά αποτελέσματα της επεξεργασίας της συνάρτησης. Οι δύο αυτοί χώροι θα δηλώνονται ως **global μεταβλητές (arrays)** και θα περνάνε σαν ορίσματα στη συνάρτηση.
- 2) Μέσω **κατάλληλων βρόγχων επανάληψης** θα γίνεται ανάγνωση των αριθμών από τη μνήμη και ο υπολογισμός της πενταπλάσιας τιμής τους. **Δεν πρέπει να χρησιμοποιήσετε την πράξη του πολλαπλασιασμού.**
- 3) Η αποθήκευση των νέων τιμών μετά την επεξεργασία θα γίνεται στο δεύτερο δεσμευμένο χώρο μνήμης.
- 4) Η συνάρτηση θα επιστρέφει τη διεύθυνση του πρώτου στοιχείου του array όπου αποθηκεύθηκαν οι νέες τιμές.
- 5) Στη main θα διαβάζονται οι νέες τιμές από τη μνήμη και θα εκτυπώνονται η μία μετά την άλλη.

#### **4. Υλοποίηση σε CLang**

1. Υλοποιήστε σε γλώσσα CLang τον παραπάνω κώδικα.
2. Χρησιμοποιήστε το NetBeans για να μεταγλωττίσετε και να εκτελέσετε την εκδοχή σε CLang, και επιβεβαιώστε την ορθή λειτουργία του προγράμματος σας.

#### **5. Υλοποίηση σε Assembly**

1. Μετατρέψτε από τη CLang σε γλώσσα Assembly : **μόνο τη δυνατότητα επιλογής του προγράμματος από το menu. Στη συνέχεια να τυπώνεται η επιλογή που κάνατε.** Το πρόγραμμα θα τρέχει επαναληπτικά μέχρι να επιλεγεί η έξοδος.
2. Εξοικειωθείτε πρώτα με τον προσομοιωτή SPIM. Κάνετε load ένα παράδειγμα .asm. Δείτε πως εκτελείται. Χρησιμοποιείτε breakpoints, και παρακολουθείστε πως αλλάζουν οι τιμές των καταχωρητών, η μνήμη κλπ. Αλλάξτε λίγο το πρόγραμμα, π.χ. δώστε διαφορετικές τιμές, και αλλάξτε τα ονόματα των labels.

#### **Παραδοτέα – Βαθμολογία**

1. Σύντομη αναφορά σχετικά με σημαντικά σημεία του κώδικα σας και τη διαδικασία υλοποίησης του εργαστηρίου – βάσει του προτύπου αναφορών.
2. Προβλήματα που αντιμετωπίσατε για μελλοντική βελτίωση του εργαστηρίου.

**ΚΑΛΗ ΕΠΙΤΥΧΙΑ**