

### 6η Σειρά Εργαστηριακών Ασκήσεων

Οι εργαστηριακές ασκήσεις είναι **ατομικές**. Οι απαντήσεις θα πρέπει να υποβληθούν με **turnin**, το αργότερο μέχρι την **Παρασκευή 14 Μαΐου 2021, ώρα 16:45**. Πριν ξεκινήσετε να γράφεται τα προγράμματα που ζητούνται στις ασκήσεις της σειράς αυτής, **διαβάστε πολύ προσεκτικά τις αναλυτικές οδηγίες** που ακολουθούν.

#### Οδηγίες

- Για τη συγγραφή των προγραμμάτων επιτρέπεται να χρησιμοποιήσετε προκαθορισμένα κατηγορήματα και προκαθορισμένους τελεστές **μόνο εφόσον αναφέρονται στις σημειώσεις του μαθήματος**.
- Για τη συγγραφή των προγραμμάτων θα πρέπει να χρησιμοποιήσετε το αρχείο πρότυπο Lab6.pro (που υπάρχει στην ιστοσελίδα του μαθήματος), στο οποίο για κάθε κατηγορήμα που ζητείτε να ορίσετε στις παρακάτω ασκήσεις, υπάρχει ένας κανόνας ο οποίος το ορίζει έτσι ώστε να επιστρέφει πάντα την απάντηση no. Για να απαντήσετε στις ασκήσεις αντικαταστήστε καθέναν από τους παραπάνω κανόνες με ένα κατάλληλο σύνολο προτάσεων που να ορίζει το αντίστοιχο κατηγορήμα. **Δεν θα πρέπει να τροποποιήσετε το όνομα κανενός κατηγορήματος ούτε το πλήθος των ορισμάτων του**.
- Μπορείτε να ορίσετε όσα βοηθητικά κατηγορήματα θέλετε, τα οποία θα χρησιμοποιούνται για τον ορισμό των κατηγορημάτων που σας ζητείται να υλοποιήσετε. Σε καμία περίπτωση δεν θα πρέπει να προσθέσετε άλλα ορίσματα στα κατηγορήματα που σας ζητούνται.
- **Αν χρησιμοποιήσετε προκαθορισμένα κατηγορήματα ή τελεστές που δεν αναφέρονται στις σημειώσεις του μαθήματος, η αντίστοιχη άσκηση δεν θα βαθμολογηθεί.**
- Ο έλεγχος της ορθότητας των απαντήσεων θα γίνει με ημι-αυτόματο τρόπο. Σε καμία περίπτωση δεν θα πρέπει ο βαθμολογητής να χρειάζεται να κάνει παρεμβάσεις στο αρχείο που θα υποβάλετε. Συνεπώς θα πρέπει να λάβετε υπόψη τα παρακάτω:
  1. Κάθε ένα από τα κατηγορήματα που σας ζητείται να υλοποιήσετε θα πρέπει να έχει το συγκεκριμένο όνομα και το συγκεκριμένο πλήθος ορισμάτων που περιγράφεται στην εκφώνηση της αντίστοιχης άσκησης και που υπάρχει στο αρχείο πρότυπο Lab6.pro. **Αν σε κάποια άσκηση το όνομα ή το πλήθος των ορισμάτων δεν συμφωνεί με αυτόν που δίνεται στην εκφώνηση, η άσκηση δεν θα βαθμολογηθεί.**
  2. Το αρχείο που θα παραδώσετε δεν θα πρέπει να περιέχει συντακτικά λάθη. Αν υπάρχουν τμήματα κώδικα που περιέχουν συντακτικά λάθη, τότε θα πρέπει να τα διορθώσετε ή να τα αφαιρέσετε πριν από την παράδοση. **Αν το αρχείο που θα υποβάλετε περιέχει συντακτικά λάθη, τότε ολόκληρη η εργαστηριακή άσκηση θα μηδενιστεί.**

3. Οι ερωτήσεις που δίνονται στο τέλος κάθε άσκησης θα πρέπει να επιστρέφουν απάντηση. Αν κάποιες από τις επιστρεφόμενες απαντήσεις δεν είναι σωστές, αυτό θα ληφθεί υπόψη στη βαθμολογία, ωστόσο η άσκηση θα βαθμολογηθεί κανονικά. **Αν ωστόσο κάποια από τις παραπάνω ερωτήσεις δεν επιστρέφει απάντηση, (π.χ. προκαλείται υπερχείλιση στοίβας, ατέρμονος υπολογισμός ή κάποιο σφάλμα χρόνου εκτέλεσης) τότε ο βαθμός για την υλοποίηση του αντίστοιχου κατηγορήματος θα είναι μηδέν.**
  4. Κατα τη διόρθωση των ασκήσεων οι βαθμολογητές δεν θα κάνουν χρησιμοποιήσουν ερωτήσεις που εμπεριέχουν τα βοηθητικά κατηγορήματα τα οποία ενδεχομένως θα έχετε ορίσει. Η χρήση των βοηθητικών κατηγορημάτων θα πρέπει να γίνεται μέσα από τα κατηγορήματα που σας ζητείται να υλοποιήσετε.
- Μετά το τέλος της εκφώνησης κάθε άσκησης δίνονται παραδείγματα ερωτήσεων με τις αντίστοιχες αναμενόμενες απαντήσεις, που μπορείτε να χρησιμοποιήσετε για έλεγχο της ορθότητας των προγραμμάτων σας.
  - Μπορείτε να συνδέεστε στην ομάδα ‘ΕΡΓΑΣΤΗΡΙΟ στις ΑΡΧΕΣ ΓΛΩΣΣΩΝ ΠΡΟΓΡΑΜΜΑΤΙΜΟΥ’ στο MsTeams τις ώρες του εργαστηρίου (Τρίτη 10-12), για την επίλυση αποριών και προβλημάτων που ενδέχεται να συναντήσετε κατά τη συγγραφή των προγραμμάτων στο πλαίσιο των εργαστηριακών ασκήσεων. Για αντίστοιχα προβλήματα ή απορίες που θα προκύψουν στο διάστημα από την περάτωση του εργαστηρίου μέχρι την υποβολή της εργασίας μπορείτε να επικοινωνήσετε με την κ. Βίκυ Σταμάτη την Παρασκευή 10:00-12:00 είτε μέσω MsTeams (προσωπικό μήνυμα) είτε μέσω ηλεκτρονικού ταχυδρομείου (vstamati@uoi.gr). Θα απαντηθούν μόνο ηλεκτρονικά μηνύματα που έχουν σταλεί από τον ιδρυματικό σας λογαριασμό.
  - Για υποβολή με turnin γράψτε:

**turnin Prolog-6@myy401 Lab6.pro**

### Άσκηση 1.

Ο διπολικός διαιρέτης ενός θετικού ακέραιου αριθμού  $n$  ορίζεται με τον παρακάτω τρόπο:

- Αν  $n = 1$  ή ο  $n$  είναι πρώτος αριθμός, τότε ο διπολικός διαιρέτης του  $n$  είναι ο εαυτός του.
- Αν ο  $n$  είναι σύνθετος αριθμός, τότε ο διπολικός διαιρέτης του  $n$  είναι το γινόμενο του μεγαλύτερου πρώτου αριθμού που διαιρεί το  $n$  επί τον μικρότερο πρώτο αριθμό που διαιρεί το  $n$ .

Γράψτε ένα πρόγραμμα σε Prolog για τον υπολογισμό του διπολικού διαιρέτη ενός δεδομένου θετικού ακέραιου αριθμού. Συγκεκριμένα, ορίστε ένα κατηγορημα `bipolarDivisor(N,D)`, το οποίο θα αληθεύει αν η τιμή της  $N$  είναι ένας θετικός ακέραιος αριθμός  $n$  και η τιμή του  $D$  είναι ο διπολικός διαιρέτης του  $n$ . Δεν μας ενδιαφέρει η συμπεριφορά του κατηγορήματος, όταν το πρώτο όρισμα δεν είναι φυσικός αριθμός.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- bipolarDivisor(14,D).  
D = 14
```

```
| ?- bipolarDivisor(31,D).  
D = 31
```

```
| ?- bipolarDivisor(81,D).  
D = 9
```

```
| ?- bipolarDivisor(303,D).  
D = 303
```

```
| ?- bipolarDivisor(308,D).  
D = 22
```

```
| ?- bipolarDivisor(525,D).  
D = 21
```

```
| ?- bipolarDivisor(1001,D).  
D = 91
```

```
| ?- bipolarDivisor(1287,D).  
D = 39
```

```
| ?- bipolarDivisor(3549,D).  
D = 39
```

```
| ?- bipolarDivisor(5005,D).  
D = 65
```

## Άσκηση 2.

Μπορούμε να αναπαραστήσουμε φυσικούς αριθμούς στην Prolog χρησιμοποιώντας το ατομικό όρο 0 και ένα συναρτησιακό σύμβολο με ένα όρισμα, π.χ. το  $s$ . Ο φυσικός αριθμός  $n$  αναπαρίσταται με τον όρο  $\underbrace{s(s(\dots s(0)\dots))}_n$  τον οποίο στη συνέχεια θα συμβολίζουμε για συντομία με  $s^n(0)$  (η μοναδιαία

αναπαράσταση περιγράφεται με περισσότερες λεπτομέρειες στις σημειώσεις, σελ 38).

Η πράξη της διαίρεσης σε μοναδιαία αναπαράσταση περιγράφεται από την παρακάτω συνάρτηση (όπου το `undefined` είναι ατομικός όρος):

$$udiv(s^n(0), s^m(0)) = \begin{cases} s^{\lfloor \frac{n}{m} \rfloor}(0) & \text{αν } m > 0 \\ \text{undefined} & \text{αν } m = 0 \end{cases}$$

Γράψτε ένα πρόγραμμα σε Prolog για υλοποίηση της διαίρεσης φυσικών αριθμών σε μοναδιαία αναπαράσταση. Συγκεκριμένα ορίστε ένα κατηγόρημα `divide(X,Y,D)`, το οποίο θα αληθεύει αν η τιμή της  $D$  ισούται με την τιμή της συνάρτησης `udiv` με ορίσματα τις τιμές των  $X, Y$ . Το πρόγραμμα δεν επιτρέπεται να μετατρέπει τους όρους που αναπαριστούν ακέραιους σε ακέραιους της Prolog και να χρησιμοποιεί τους ενσωματωμένους αριθμητικούς τελεστές και το προκαθορισμένο κατηγόρημα `is` για αποτίμηση αριθμητικών παραστάσεων.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- divide(s(0),s(0),D).
D = s(0) ?
| ?- divide(s(s(s(s(0))))),s(0),D).
D = s(s(s(s(0))))
| ?- divide(s(s(s(s(s(s(s(s(0))))))))) ,s(s(0)),D).
D = s(s(s(s(0))))
| ?- divide(s(s(s(s(s(s(s(s(s(0))))))))) ,s(s(s(0))),D).
D = s(s(0))
| ?- divide(s(s(s(s(s(s(s(s(s(0))))))))) ,s(s(s(s(0))))),D).
D = s(s(0))
| ?- divide(s(s(s(s(0))))),s(s(s(s(0))))),D).
D = s(0)
| ?- divide(s(s(s(s(s(0))))),s(s(s(s(0))))),D).
D = s(0)
| ?- divide(s(s(s((0))))),s(s(s(s(0))))),D).
D = 0
| ?- divide(s(s(s((0))))),0,D).
D = undefined
| ?- divide(0,s(s(s((0))))),D).
D = 0
```