# ΠΛΗΡΟΦΟΡΙΚΗ Ι
# ΕΡΓΑΣΤΗΡΙΟ 3

## Θέμα: Επαναληπτικές διαδικασίες

**1.** Να γραφεί πρόγραμμα `fact` που να δέχεται θετικό ακέραιο και να υπολογίζει το παραγοντικό του.

**2.** Να γραφεί πρόγραμμα που να εμφανίζει τους περιττούς αριθμούς μεταξύ του 1 και δοσμένου αριθμού. Να γίνουν *δύο* εκδόσεις του προγράμματος, μία με `for (odd1.py)` και μία με `while (odd2.py)`.

**3.** Να γραφεί πρόγραμμα `mesos_oros1` που να υπολογίζει τον μέσο όρο των άρτιων από γνωστό πλήθος θετικών αριθμών. Το πλήθος καθώς και οι ίδιοι οι αριθμοί θα δίνονται από τον χρήστη.

**4.** Τροποποιήστε το προηγούμενο πρόγραμμα (`mesos_oros2`) ώστε να κάνει το ίδιο για άγνωστο πλήθος θετικών αριθμών (ο τερματισμός της διαδικασίας εισόδου αριθμών να γίνεται με την είσοδο οποιουδήποτε αρνητικού αριθμού).

**5.** Το παρακάτω πρόγραμμα ελέγχει εάν η είσοδός του είναι πρώτος αριθμός και εμφανίζει σχετικό μήνυμα:

```
n=0
while n<2:
    n = int(input('Δώσε ακέραιο μεγαλύτερο του 1: '))
i=2
while n%i != 0:
    i += 1
if n==i:
   print('Ο', n, 'είναι πρώτος')
else:
   print('Ο', n, 'δεν είναι πρώτος')
```

α) Ποιος είναι ο ρόλος του πρώτου `while` στο πρόγραμμα αυτό;
β) Ποιος είναι ο αλγόριθμος βάσει του οποίου προκύπτει εάν η είσοδος είναι πρώτος ή όχι;
γ) Τροποποιήστε το πρόγραμμα ώστε να χρησιμοποιεί `for` αντί για `while`.
δ) Τροποποιήστε το πρόγραμμα ώστε να τρέχει πιο γρήγορα.

**6.** Να γραφεί πρόγραμμα `epsilon` που να ζητάει από τον χρήστη την είσοδο ενός βαθμού ακρίβειας ώστε να προσεγγίσει την τιμή του $e$ με τη συγκεκριμένη ακρίβεια. Η τιμή του $e$ μπορεί να προσεγγιστεί ως εξής:

$$e = \sum_{k=0}^{\infty} \frac{1}{k!} = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \cdots + \frac{1}{n!} + \cdots$$

και το σφάλμα προσέγγισης μετά από $n$ όρους είναι:

$$\text{σφάλμα} \leq \frac{3}{(n+1)!} \tag{1}$$

*Στο πρόγραμμα να μη χρησιμοποιηθεί η πραγματική τιμή του $e$ για σύγκριση με την προσέγγιση, αλλά να χρησιμοποιηθεί η ανισότητα (1) ώστε να τερματίζεται ο αλγόριθμος όταν το σφάλμα γίνεται μικρότερο της ακρίβειας που δίνει ο χρήστης. Επίσης, να χρησιμοποιηθεί το ότι $(n+1)! = n! \cdot (n+1)$*

**7.** Να γραφεί πρόγραμμα `million` που να υπολογίζει τον μικρότερο θετικό ακέραιο αριθμό του οποίου το παραγοντικό έχει τουλάχιστον 1 εκατομμύριο ψηφία.

Χρησιμοποιήστε τα εξής δεδομένα:

(i) Η εντολή `math.floor(math.log10(x))+1` δίνει το πλήθος των ψηφίων του `x`, όπου `x` θετικός ακέραιος.

(ii) $\log_{10}(n!) = \log_{10}(2) + \log_{10}(3) + \cdots + \log_{10}(n)$

*(Θα χρειαστείτε τη `while` για τις επαναλήψεις καθώς και ένα τρέχον σύνολο για τα αθροίσματα των λογαρίθμων.)*

**8.** Να γραφεί πρόγραμμα `stars` που να εκτυπώνει στην οθόνη το παρακάτω τρίγωνο από αστερίσκους. Το πλήθος των αστερίσκων της διαγωνίου $(n)$ να δίνεται ως είσοδος από τον χρήστη.

```
        *          γραμμή 1: n-1 κενά, *
      *   *        γραμμή 2: n-2 κενά, *, 0 κενά , *
    *       *      γραμμή 3: n-3 κενά, *, 1 κενό , *
  *           *    γραμμή 4: n-4 κενά, *, 2 κενά , *
* * * * * * * * *  γραμμή 5: n αστερίσκοι
```