
Ασκήσεις μελέτης Analysis- Lab 2

1

1.1 Uniform numbers - uninum.py

- Για την παραγωγή τυχαίων αριθμών στο $[0, 1)$ μπορούμε να χρησιμοποιήσουμε την συνάρτηση `np.random.random()`
- Για την παραγωγή τυχαίων αριθμών στο $[a, b]$ θα το απαντήσουμε με τα ακόλουθα 2 βήματα. Αρχικά έστω ότι θέλουμε να έχουμε τους αριθμούς στο διάστημα $[0, x)$. Αρκεί να κάνουμε **scaling** τα άκρα κατά μία μεταβλητή x , δηλαδή $x * np.random.random()$. Τώρα για να φέρουμε το $[0, x)$ στο $[a, b]$ αρκεί να κάνουμε **shift** τα δύο άκρα κατά μία μεταβλητή y , δηλ $[y, x + y)$. Οπότε έχουμε το σύστημα
 $a = y$ και $x + y = b$, οπότε καταλήγουμε
 $y = a$ και $x = b - a$
και συνεπώς
 $a + (b - a) * np.random.random()$
- Για την παραγωγή τυχαίων αριθμών σε ένα ορθογώνιο $[a, b] \times [c, d]$ ακολουθούμε ανεξάρτητα την προηγούμενη διαδικασία για κάθε διάστημα.
- Για την παραγωγή τυχαίων ακέραιων αριθμών στο διάστημα $[n, m]$, ας απαντήσουμε το ερώτημα αρχικά για το διάστημα ακέραιων $[0, e]$. Κόβουμε τον χώρο $[0, 1)$ σε (k) ίσα διαστήματα. Έστω $step = 1/k$, για να απαντήσουμε σε ποιο διάστημα ανήκει ένας τυχαίος αριθμός $x \in [0, 1)$ αρκεί να βρούμε ένα i τέτοιο ώστε

$$step \times i \leq x \leq step \times (i + 1)$$

που δεν είναι τίποτα άλλο από το $i = \lfloor \frac{x}{step} \rfloor$

Ακολουθώντας την ίδια διαδικασία με πριν μπορούμε να κάνουμε **shift** ώστε να φέρουμε το διάστημα στο $[n, m]$

- Εναλλακτικά μπορούμε να παράξουμε πραγματικούς αριθμούς στο διάστημα $[n, m + 1)$ και να κρατήσουμε το ακέραιο μέρος τους

1.2 Triangular Matrix - trimax.py

Για να απαντήσουμε αυτό το ερώτημα πρέπει να χρησιμοποιήσουμε και τον ανάστροφο πίνακα του A (τον οποίο μπορείτε να τον σκεφτείτε ως reflection γύρω από την κύρια διαγώνιο).

Έστω $c1 = isUpperTrig(A)$

και

$c2 = isUpperTrig(tr(A))$

Αν $not_rect(A)$ τότε -1

Αλλιώς_Αν $c1$ και $c2$ αληθή τότε είναι διαγώνιος

Αλλιώς_Αν $c1$ αληθές τότε άνω τριγωνικός

Αλλιώς_Αν $c2$ αληθές τότε κάτω τριγωνικός

Αλλιώς τετραγωνικός