

Πρώτη εργαστηριακή άσκηση στο μάθημα Αρχές Γλωσσών Προγραμματισμού

Τμήμα Πληροφορικής και Τηλεπικοινωνιών, Πανεπιστημίου Ιωαννίνων
https://chggogos.github.io/dituoi_ARCHES_GLOSSON_PROGRAMMATISMOU/

Άρτα, 15 Φεβρουαρίου 2025

Εισαγωγή

Η εργαστηριακή άσκηση εξετάζει θέματα προγραμματισμού με την Python όπως οι λίστες και ο χειρισμός τους (π.χ. slicing), η δημιουργία τυχαίων τιμών, ο ορισμός και η κλήση συναρτήσεων, η κλήση συναρτήσεων από modules της standard βιβλιοθήκης, ο διαμοιρασμός κώδικα σε modules, η ανάγνωση αρχείων κειμένου, τα unit tests, οι κανονικές εκφράσεις, η χρονομέτρηση κώδικα και άλλα.

Περιγραφή

Στην άσκηση αυτή ζητείται ο χειρισμός αρχείων κειμένου που περιέχουν εντολές πολλαπλασιασμού της μορφής ΠΟΛΛΑΠΛΑΣΙΑΣΕ(x, y) όπου x και y είναι ακέραιοι αριθμοί.

Ερώτημα 1

Δημιουργήστε ένα module με όνομα `erotima1.py` που θα περιέχει τις ακόλουθες συναρτήσεις:

1. `generate_random_text(n)`
2. `inject_multiplications(txt, n)`
3. `compute(fn)`

Η συνάρτηση `generate_random_text(n)`: Η συνάρτηση `generate_random_text(n)` θα δημιουργεί ένα κείμενο με n τυχαίους χαρακτήρες από το ελληνικό αλφάβητο (πεζά και κεφαλαία) και τα ψηφία 0-9. Για παράδειγμα η κλήση της συνάρτησης με όρισμα την τιμή 100 θα μπορούσε να επιστρέψει ένα κείμενο όπως το ακόλουθο:

```
μΖσ5ηΜκΑΩανΟΕαβΗΓΘ1ΞαΗγΔπΒμιΕ19Ν1ΝΒχΟ22ρβζΨκΒοΘλΝΞ  
ΘΠοΓΘ2ΙμΩΖξζ0Σε4νΜ4ψΦκελ33ζβφε0ωνδσ9ξΒ0Χ6π5ΠΨΟλκοΥ
```

1: Παράδειγμα κειμένου που μπορεί να παράξει η `generate_random_text(300)`

Η συνάρτηση `inject_multiplications(txt, n)`: Η συνάρτηση `inject_multiplications(txt, n)` θα δέχεται ως όρισμα ένα κείμενο `txt` και μια ακέραια τιμή `n`. Θα επιστρέφει ένα νέο κείμενο που θα προκύπτει από το αρχικό, στο οποίο θα έχει εισαχθεί `n` φορές, σε τυχαίες θέσεις αντικαθιστώντας το υπάρχον κείμενο, κείμενο της μορφής ΠΟΛΛΑΠΛΑΣΙΑΣΕ(x, y) όπου x και y είναι ακέραιοι αριθμοί από το 0 μέχρι και το 99. Το μήκος του αρχικού και του τελικού κειμένου θα πρέπει να είναι ίσα μεταξύ τους. Να υπάρχει assertion που να ελέγχει ότι το όρισμα `txt` περιέχει κείμενο με μήκος μεγαλύτερο από 20 χαρακτήρες. Η κλήση της `inject_multiplications(txt, n)` με ορίσματα το προηγούμενο κείμενο και την τιμή 3 θα μπορούσε να επιστρέψει ένα κείμενο όπως το ακόλουθο:

```
μΖσ5ηΜκΑΩΠΟΛΛΑΠΛΑΣΙΑΣΕ(37,46)19Ν1ΝΒΠΟΛΛΑΠΛΑΣΙΑΣΕ(7  
1,18)(20,35)0Σε4νΜ4ψΦκελ33ζβφε0ωνδσ9ξΒ0Χ6π5ΠΨΟλκοΥ
```

2: Παράδειγμα κειμένου που παράγει η συνάρτηση `inject_multiplications(...)`

Παρατηρήστε ότι είναι πιθανό οι διαδοχικές εισαγωγές του κειμένου ΠΟΛΛΑΠΛΑΣΙΑΣΕ(. . .) να επικαλύπτουν τμήματα προηγούμενων εισαγωγών. Αυτό συμβαίνει στο παραπάνω παράδειγμα όπου οι εισαγωγές κειμένου είναι οι ακόλουθες:

ΠΟΛΛΑΠΛΑΣΙΑΣΕ(20,35) εισαγωγή στη θέση 42

ΠΟΛΛΑΠΛΑΣΙΑΣΕ(37,46) εισαγωγή στη θέση 9

ΠΟΛΛΑΠΛΑΣΙΑΣΕ(71,18) εισαγωγή στη θέση 35

Η συνάρτηση compute(txt): Η συνάρτηση compute(txt) θα εξετάζει το κείμενο txt και θα υπολογίζει το άθροισμα από όλα τα γινόμενα από έγκυρες εμφανίσεις του ΠΟΛΛΑΠΛΑΣΙΑΣΕ(x,y) όπου x και y είναι ακέραιοι αριθμοί. Για την υλοποίηση της συνάρτησης να χρησιμοποιηθεί η βιβλιοθήκη re¹ της Python. Για παράδειγμα η κλήση της συνάρτησης για το παραπάνω κείμενο θα πρέπει να επιστρέψει την τιμή 2980 που ισούται με $37*46 + 71*18$.

Ερώτημα 2

Σε αρχείο με όνομα erotima2.py προσθέστε unit test που να ελέγχει ότι η compute(txt) παράγει ορθά αποτελέσματα για 5 περιπτώσεις διαφορετικής εισόδου δικής σας επιλογής. Προσθέστε ένα επιπλέον unit test που να ελέγχει ότι η compute(txt) παράγει το σωστό αποτέλεσμα (που είναι 1035105) για τα περιεχόμενα του αρχείου test_multiplications1.txt². Μπορείτε να χρησιμοποιήσετε ως testing framework, είτε το unittest³, είτε το pytest⁴.

Ερώτημα 3

Γράψτε κώδικα σε αρχείο με όνομα erotima3.py που:

1. να δημιουργεί ένα κείμενο με 1000000 χαρακτήρες με τη συνάρτηση generate_random_text(...),
2. να εισάγει στο κείμενο 100000 εντολές πολλαπλασιασμού της μορφής ΠΟΛΛΑΠΛΑΣΙΑΣΕ(x,y) με τη συνάρτηση inject_multiplications(...),
3. να καλεί τη συνάρτηση compute(...) για το νέο κείμενο και να εμφανίζει το αποτέλεσμα, δηλαδή το άθροισμα από όλους τους πολλαπλασιασμούς που εντοπίζει στο κείμενο και
4. να εμφανίζει το χρόνο εκτέλεσης που χρειάστηκε καθένα από τα 3 προηγούμενα βήματα.

Υποβολή και εξέταση της εργαστηριακής άσκησης

Η προθεσμία υποβολής της εργαστηριακής άσκησης είναι στις 10/3/2025 (Δευτέρα). Στις 11/3/2025 (Τρίτη) και 13/3/2025 (Πέμπτη) στο εργαστήριο του κάθε τμήματος, οι φοιτητές που θα έχουν υποβάλει την εργαστηριακή άσκηση θα εξεταστούν γραπτώς σε θέματα κώδικα παρόμοια με αυτά που χρειάστηκε να αναπτύξουν για την υλοποίηση της εργαστηριακής άσκησης. Η μη συμμετοχή στην εξέταση της εργαστηριακής άσκησης ακυρώνει την υποβολή της. Επιπλέον, ισχύουν τα ακόλουθα:

- Η εργασία μπορεί να υποβληθεί μόνο στο <https://ecourse.uoi.gr/course/view.php?id=1945>.
- Η εργασία είναι ατομική και η γλώσσα υλοποίησης είναι η Python.
- Θα πρέπει να έχει εφαρμοστεί κάποιος code formatter σε όλο τον κώδικα (π.χ. black⁵).
- Τα παραδοτέα (σε ένα zip αρχείο με όνομα AM_EPVNYMO_ONOMA.zip) όπου AM είναι ο αριθμός μητρώου του φοιτητή που υποβάλει την άσκηση, ενώ το επώνυμο και το όνομα θα πρέπει γραφούν με λατινικούς χαρακτήρες, είναι τα ακόλουθα:
 - Κώδικας (τα αρχεία erotima1.py, erotima2.py, erotima3.py).
 - Αρχείο README.txt με οδηγίες εκτέλεσης και τους χρόνους εκτέλεσης που ζητούνται στο ερώτημα 3.

¹<https://docs.python.org/3/library/re.html>

²Κατεβάστε το αρχείο test_multiplications1.txt από το <https://tinyurl.com/5x4w9wcm>

³<https://docs.python.org/3/library/unittest.html>

⁴<https://docs.pytest.org/en/stable/>

⁵<https://github.com/psf/black>