ΟΔΗΓΙΕΣ ΕΚΤΕΛΕΣΗΣ

Το αρχείο 2565\_final\_project.py περιλαμβάνει την υλοποίηση όλης της άσκησης. Οπότε μόνο αυτό το αρχείο χρειάζεται να τρέξει. Οι σειρές 1-58 αφορούν τα ερωτήματα 1-2, οι 60-172 το ερώτημα 3 και οι τελευταίες το ερώτημα 4.

Για όλα τα ερωτήματα υπάρχει και ξεχωριστός κώδικας τους.

Ο κώδικας που αφορά το πρώτο μέρος της εργασίας υλοποιείται με την χρήση dictionaries (υπάρχει και ξεχωριστός ο κώδικας στο αρχείο 1-2\_dictionary.py) και τυπώνει τα αντίστοιχα δεδομένα. Υπάρχει και το αρχείο 1-2\_data\_frame.py το οποίο παρουσιάζει μια διαφορετική λύση με την χρήση pandas και Data Frame.

Το ερώτημα 3 υλοποιείται στον κώδικα κανονικά και υπάρχει και το αντίστοιχό αρχείο 3.py που εμφανίζεται μόνος του ο αντίστοιχος κώδικας και η δοκιμή του με ενδεικτικό unittest.

Το ερώτημα 4 για 1.000.000 χρεώσεις τρέχει πολύ αργά και έβαλα τον κώδικα σαν default να τρέχει για 10.000 όπου τυπώνει κανονικά τα αποτελέσματα. Όσο αυξάνεται το πλήθος των χρεώσεων τόσο πιο αργή είναι η εκτέλεση του κώδικα.

Το πρόβλημα είναι πως η υλοποίηση με dictionary και αυτή με hash table εμφανίζουν διαφορετικά αποτελέσματα για τον ίδιο αριθμό χρεώσεων.

Επίσης υπάρχει ανεβασμένη τεχνική αφορά στο περιβάλλον του Latex μαζί με τον αντίστοιχο κώδικα.

Όλα τα αρχεία υπάρχουν και στο:

https://github.com/konrantos/alco/tree/main/project1

ΠΗΓΕΣ

<https://eng.libretexts.org/Courses/Folsom_Lake_College/CISP_430%3A_Data_Structures_(Aljuboori)/07%3A_Hash_Tables/7.02%3A_Choosing_a_good_hash_function>

<https://stackoverflow.com/questions/34264710/what-is-the-point-of-floatinf-in-python>