Τεχνητή Νοημοσύνη

Project 0
Κωνσταντίνος Πυθαρούλιος
A.M.: 111520 1600142

addition.py

Η συνάρτηση add είναι μια πολύ απλή συνάρτηση που δέχεται δύο αριθμούς και υπολογίζει το άθροισμά τους.

buyLotsOfFruit.py

Η συνάρτηση buyLotsOfFruits παίρνει ως είσοδο μια λίστα από πλειάδες, με ονόματα φρούτων και την ποσότητα που θέλουμε από το καθένα. Αφότου υπολογίσει το κόστος της ποσότητας κάθε φρούτου, επιστρέφει το συνολικό κόστος όλης της παραγγελίας.

Αξίζει εδώ να σημειωθεί, πως δεν γίνεται import του shop.py οπότε δεν μπορούμε να χρησιμοποιήσουμε κάποια έτοιμη συνάρτηση από αυτό το αρχείο. Για αυτό και δημιουργούμε τη δική μας συνάρτηση getFruitCostPart(fruit) που επιστρέφει την τιμή της μονάδας κάθε φρούτου που επιθυμούμε, ώστε να μας βοηθήσει στον υπολογισμό του συνολικού κόστους κάθε φρούτου και σενεπώς όλης της παραγγελίας.

shopSmart.py

Η συνάρτηση shopSmart παίρνει ως όρισμα μια λίστα από πλειαδες, με ονόματα φρούτων και την επιθυμητή ποσότητά τους, και μια λίστα από fruitShops. Τελικά, επιστρέφει το fruitShop που προσφέρει την παραγγελία στην χαμηλώτερη τιμή.
Εδώ, γίνεται import shop, οπότε χρησιμοποιούμε στο εσωτερικό της συνάρτησής μας, την έτοιμη συνάρτηση *getPriceOfOrder(orderList), που υπολογίζει την συνολική τιμή μιας παραγγελίας για ένα fruitShop.*

priorityQueue.py

Καταρχήν, νομίζω πως το όνομα που μας ζητήθηκε να δώσουμε στο αρχείο μας δεν αντικατοπτρίζει ακριβώς το περιεχόμενο του και αυτό γιατί στο εσωτερικό του υλοποιούμε και χρησιμοποιούμε μια στοίβα (stack). Αν ήταν priority queue θα έπρεπε να χρησιμοποιούνται βάρη που θα δείχουν το priority αλλά και η μεθοδολογία να είναι fifo.
Αντίθετα εμείς χρησιμοποιούμε, όπως αναφέραμε, μια Stack με την τυπική της μέθοδο, τη lifo.

Υλοποιούνται όλες οι βασικές της λειτουργίες. Δηλαδή το initialization, ο έλεγχος αν είναι άδεια, ο έλεγχος αν είναι γεμάτη, push και pop.
Για τον έλεγχο του αν η στοίβα είναι γεμάτη, πρέπει να σημειωθεί πως χρησιμοποιούμε μια μεταβλητή MAXSTACKSIZE που είναι ίση με 100. Δηλαδή θεωρούμε πως η στοίβα χωράει μέχρι 100 items.

Κατά την εκτέλεση, το πρόγραμμα ζητάει από το χρήστη να δώσει ένα string με παρενθέσεις και έπειτα αυτό ελέγχεται για να απαντηθεί αν τελικά ήταν ισορροπημένο ως προς αυτές ή όχι.
Η μέθοδος που χρησιμοποιήθηκε ήταν η ίδια με την αντίστοιχη που εμφανίζεται στις διαφάνεις που μας δόθηκαν και ήταν γραμμένη σε γλώσσα C.

Η λογική είναι απλή. Αν διαβάζεται μια παρένθεση που ανοίγει μπαίνει στη στοίβα. Μόλις διαβαστεί μια που κλείνει, βγάζουμε το τελευταίο στοιχείο που εισήλθε στη στοίβα και αυτό θα πρέπει να είναι το ίδιο σύμβολο απλά τώρα να ανοίγει και όχι να κλείνει.