

Ε.Α.Π./ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

3η ΓΡΑΠΤΗ ΕΡΓΑΣΙΑ

ΑΚΑΔΗΜΑΪΚΟΥ ΕΤΟΥΣ 2021-2022

21/01/2022

Ημερομηνία παράδοσης εργασίας: Κυριακή 06/03/2022

Καταληκτική ημερομηνία παραλαβής: Τετάρτη ¹ 09/03/2022

Ημερομηνία ανάρτησης ενδεικτικών λύσεων: Σάββατο 12/03/2022

Καταληκτική ημερομηνία αποστολής σχολίων στον φοιτητή: Κυριακή 27/03/2022

ΥΠΟΕΡΓΑΣΙΑ 1.

(βαθμοί 25)

Εισαγωγή στον Αντικειμενοστραφή Προγραμματισμό

ΥΠΟΕΡΓΑΣΙΑ 2.

(βαθμοί 25)

Κλάσεις και Αντικείμενο

ΥΠΟΕΡΓΑΣΙΑ 3.

(βαθμοί 25)

Μέθοδοι οριζόμενοι από τον χρήστη

ΥΠΟΕΡΓΑΣΙΑ 4.

(βαθμοί 25)

Κληρονομικότητα και Πολυμορφισμός

ΣΥΝΟΛΟ

(βαθμοί 100)

¹ Σύμφωνα με τον Κανονισμό Σπουδών, η καταληκτική ημερομηνία για την παραλαβή της Γ.Ε. από το μέλος ΣΕΠ είναι η επόμενη Τετάρτη από το τέλος της εβδομάδας παράδοσης Γ.Ε.

ΥΠΟΕΡΓΑΣΙΑ 1.

(βαθμοί 25)

Ζητείται να συμπληρώσετε το πρόγραμμα που περιέχεται στο αρχείο `1_code_template.py` το οποίο περιέχει δύο κλάσεις, την κλάση `Stack()`, που υλοποιεί μια στοίβα (ερώτημα (α)), με τις μεθόδους της και την κλάση `Control()` που διαχειρίζεται την ώθηση(εισαγωγή)/απόθεση(εξαγωγή)/παρουσίαση (ερωτήματα (β), (γ) και (δ)) των στοιχείων της στοίβας. Κατά την απόθεση να εμφανίζεται κατάλληλο μήνυμα.

Στοίβα είναι μια δομή δεδομένων, όπου το εισερχόμενο στοιχείο ωθείται (προστίθεται) στην κορυφή (*push*), ενώ το εξερχόμενο απωθείται (αφαιρείται) από την κορυφή (*pop*).

Μια τυπική χρήση του κώδικα φαίνεται στη συνέχεια:

```
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:+1
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:+2
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:+3
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:+4
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:?
->      4
        3
        2
        1
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:-
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:?
->      3
        2
        1
+item για ώθηση, - για απώθηση, ? για παρουσίαση, x για έξοδο:x
|
```

Υπόδειξη: Να χρησιμοποιηθεί το αρχείο `1_code_template.py` ως οδηγός επίλυσης.

ΥΠΟΕΡΓΑΣΙΑ 2.

(βαθμοί 25)

Ζητείται να συμπληρώσετε το πρόγραμμα που περιέχεται στο αρχείο `2_code_template.py` το οποίο περιέχει δύο κλάσεις, την κλάση `Student()` που περιέχει αντικείμενα με τα στοιχεία φοιτητών και την κλάση `Control()` που διαχειρίζεται τη διεπαφή με τον χρήστη η οποία επιτρέπει την εισαγωγή/διαγραφή/παρουσίαση των στοιχείων των φοιτητών.

Ζητείται να συμπληρωθεί στα κατάλληλα σημεία ο κώδικας ώστε να υποστηρίζονται οι εξής λειτουργίες:

(α) Όταν ο χρήστης πατήσει "+" το πρόγραμμα ζητάει από τον χρήστη τα στοιχεία ενός νέου φοιτητή (όνομα, αριθμός μητρώου, βαθμός) χωρισμένα με κόμματα. Εάν ο χρήστης δώσει σωστά τα δεδομένα, δημιουργεί ένα νέο αντικείμενο της κλάσης `Student` και το αποθηκεύει

σε ένα λεξικό της κλάσης `Control` που ονομάζεται `theStudents`. Η λειτουργία αυτή υλοποιείται από τη μέθοδο `add_student()` την οποία πρέπει να συμπληρώσετε.

Προσοχή: Επειδή οι αριθμοί μητρώου είναι μοναδικοί (κλειδιά), αν ο αριθμός μητρώου υπάρχει ήδη, ρωτάμε τον χρήστη αν θέλει να αλλάξει τα στοιχεία του υπάρχοντος φοιτητή.

(β) Όταν ο χρήστης πατήσει "x" το πρόγραμμα ζητάει τον αριθμό μητρώου του φοιτητή και στη συνέχεια τον διαγράφει. Η λειτουργία αυτή υλοποιείται από τη μέθοδο `delete_student()` την οποία πρέπει να συμπληρώσετε.

(γ) Όταν ο χρήστης πατήσει "?" το πρόγραμμα τυπώνει όλους τους φοιτητές κατά αύξοντα αριθμό μητρώου. Η λειτουργία αυτή υλοποιείται από τη μέθοδο `show_students()` την οποία πρέπει να συμπληρώσετε.

Χρησιμοποιήστε αμυντικό προγραμματισμό όπου απαιτείται.

Υπόδειγμα χρήσης:

Προσθήκη φοιτητή (+), διαγραφή φοιτητή (x), εμφάνιση φοιτητών
(?), <enter> για έξοδο

...+

δώσε στοιχεία φοιτητή AM,ONOMA,BAΘΜΟΣ (enter για τέλος): 050,
Maria, 8

Προστέθηκε ένας φοιτητής, συνολικό πλήθος: 1

δώσε στοιχεία φοιτητή AM,ONOMA,BAΘΜΟΣ (enter για τέλος): 030,
Kostas, 10

Προστέθηκε ένας φοιτητής, συνολικό πλήθος: 2

δώσε στοιχεία φοιτητή AM,ONOMA,BAΘΜΟΣ: (enter για τέλος)

Προσθήκη φοιτητή (+), διαγραφή φοιτητή (x), εμφάνιση φοιτητών
(?), <enter> για έξοδο

...?

[030] Kostas 10.0

[050] Maria 8.0

Προσθήκη φοιτητή (+), διαγραφή φοιτητή (x), εμφάνιση φοιτητών
(?), <enter> για έξοδο

...

Υπόδειξη: Να χρησιμοποιηθεί το αρχείο `2_code_template.py` ως οδηγός επίλυσης.

ΥΠΟΕΡΓΑΣΙΑ 3.

(βαθμοί 25)

Η άσκηση έχει ως στόχο την δημιουργία ενός προγράμματος υπολογισμού της συνολικής απόστασης που θα διανύσει ένας ταξιδιώτης ο οποίος πραγματοποιεί ένα ταξίδι με ενδιάμεσες πτήσεις ανταπόκρισης. Κάθε αεροδρόμιο χαρακτηρίζεται από το όνομά του και τις συντεταγμένες του.

Δίνεται ένας σκελετός του προγράμματος (αρχείο `3_code_template.py`) που περιλαμβάνει τις κλάσεις Αεροδρόμιο (`Airport()`), Ταξίδι (`Trip()`) και `Menu()`. Οι κλάσεις διαθέτουν μέθοδο αρχικοποίησης (`__init__`) και μέθοδο εκτύπωσης (`__str__`) των αντικειμένων της κλάσης. Επιπλέον, η κλάση Αεροδρόμιο διαθέτει τη μέθοδο `get_distance(self, other)` η οποία επιστρέφει την απόσταση ανάμεσα στο αεροδρόμιο και το αεροδρόμιο `other`.

Αφού μελετήσετε τον κώδικα που δίνεται, ζητείται να τον συμπληρώσετε ώστε να έχει την εξής συμπεριφορά:

Παρακαλώ εισάγετε δρομολόγιο ως ακολουθία κωδικών αεροδρομίων,
πχ. `KLX-ATH-RHO-ATH`

Διαθέσιμα αεροδρόμια: `ATH, JTR, KLX, RHO, SKG`

>>**SKG-ATH-SKG**

Υπολογισμός αποστάσεων πτήσης

<Thessaloniki, Athens> 299.40

<Athens, Thessaloniki> 299.40

Συνολική απόσταση: 598.80

Παρακαλώ εισάγετε δρομολόγιο ως ακολουθία κωδικών αεροδρομίων,
πχ. `KLX-ATH-RHO-ATH`

Διαθέσιμα αεροδρόμια: `ATH, JTR, KLX, RHO, SKG`

>>

Να απαντήσετε στα παρακάτω ερωτήματα χωρίς να τροποποιήσετε τον σκελετό του προγράμματος:

(α) Να συμπληρωθεί η μέθοδος (`__str__`) της κλάσης `Trip()` η οποία επιστρέφει ως συμβολοσειρά την περιγραφή της πτήσης σύμφωνα με το παραπάνω παράδειγμα εκτέλεσης του προγράμματος.

(β) Να συμπληρωθεί η μέθοδος-κατασκευαστής (`__init__`) της κλάσης `Menu()` ώστε με βάση τα δεδομένα που έδωσε ο χρήστης να παράγεται ένα αντικείμενο `Trip` και στη συνέχεια να το τυπώνει με χρήση της μεθόδου (`__str__`) της κλάσης των πτήσεων.

Υπόδειξη: Να χρησιμοποιηθεί το αρχείο `3_code_template.py` ως οδηγός επίλυσης.

ΥΠΟΕΡΓΑΣΙΑ 4.

(βαθμοί 25)

Μια μικρή μεταφορική εταιρεία της Πάτρας εκτελεί για λογαριασμό των ΕΛΤΑ, τα δρομολόγια Πάτρα - Αθήνα - Θεσσαλονίκη, μεταφέροντας δέματα μεταξύ των αντίστοιχων κέντρων διαλογής. Σε κάθε δρομολόγιο η μεταφορική εταιρεία φορτώνει δέματα από το κέντρο διαλογής Πάτρας και τα μεταφέρει στην Αθήνα ή Θεσσαλονίκη.

Κάθε δέμα (Package) χαρακτηρίζεται από τον προορισμό του (Αθήνα ή Θεσσαλονίκη) και υπάγεται σε μία κατηγορία. Ανάλογα με την κατηγορία του, χαρακτηρίζεται από διαφορετικές ιδιότητες βάσει των οποίων υπολογίζεται το κόστος για τη μεταφορά του. Οι κατηγορίες δεμάτων είναι οι ακόλουθες:

- **Πακέτο (Parcel):** δέμα μικρού όγκου το οποίο χρεώνεται με βάση το βάρος του σε κιλά. Το κόστος μεταφοράς ανέρχεται σε 50 λεπτά ανά κιλό.
- **Φάκελος (Envelope):** απλός φάκελος με έγγραφα ο οποίος μπορεί να παραδοθεί με διαφορετικές προτεραιότητες. Η προτεραιότητα (priority) λαμβάνει τιμές 1 – 3, με την υψηλή προτεραιότητα (3) να αντιστοιχεί σε αυθημερόν παράδοση. Το κόστος μεταφοράς υπολογίζεται ως το γινόμενο της προτεραιότητας επί 20 λεπτά.
- **Ογκώδες αντικείμενο (Bulky_Item):** αντικείμενο το οποίο έχει μεγάλο όγκο και χαρακτηρίζεται από τις διαστάσεις του: μήκος (length), πλάτος (width), ύψος (height). Το κόστος μεταφοράς υπολογίζεται ως το γινόμενο του όγκου του σε m^3 επί 20 ευρώ ανά m^3 .

Στο κόστος μεταφοράς κάθε δέματος προστίθεται ένα ποσό το οποίο εξαρτάται από τον προορισμό του και ανέρχεται σε 1 ευρώ για την Αθήνα και 2 ευρώ για τη Θεσσαλονίκη.

Ζητείται να συμπληρώσετε το πρόγραμμα που περιέχεται στο αρχείο `4_code_template.py`. Συγκεκριμένα, να συμπληρώσετε τις κλάσεις `Parcel()`, `Envelope()`, `Bulky_Item()`, οι οποίες κληρονομούν τις ιδιότητες της υπερκλάσης `Package()` και εξειδικεύουν τις μεθόδους τους `cost()` και `__str__()` ώστε το πρόγραμμα να παράγει το παρακάτω αποτέλεσμα:

Δελτίο αποστολής

Πακέτο1 : Προορισμός: Αθήνα, Βάρος: 20.0kg

Πακέτο2 : Προορισμός: Θεσσαλονίκη, Βάρος: 10.0kg

Πακέτο3 : Προορισμός: Αθήνα, Βάρος: 30.0kg

Πακέτο4 : Προορισμός: Αθήνα, Βάρος: 4.5kg

Φάκελος1 : Προορισμός: Θεσσαλονίκη, : Προτεραιότητα 1

Φάκελος2 : Προορισμός: Θεσσαλονίκη, : Προτεραιότητα 3

ΟγκώδεςΑντικείμενο1 : Προορισμός: Αθήνα, : Διαστάσεις 0.7 x 0.5 x 1.0m

ΟγκώδεςΑντικείμενο2 : Προορισμός: Θεσσαλονίκη, : Διαστάσεις 1.0 x 0.5 x 1.0m

ΟγκώδεςΑντικείμενο3 : Προορισμός: Αθήνα, : Διαστάσεις 2.0 x 0.7 x 0.7m

Τιμολόγιο

Πακέτο1 : Προορισμός: Αθήνα, Βάρος: 20.0kg, Κόστος: 11.00€

Πακέτο2 : Προορισμός: Θεσσαλονίκη, Βάρος: 10.0kg, Κόστος: 7.00€

Πακέτο3 : Προορισμός: Αθήνα, Βάρος: 30.0kg, Κόστος: 16.00€

Πακέτο4 : Προορισμός: Αθήνα, Βάρος: 4.5kg, Κόστος: 3.25€

Φάκελος1 : Προορισμός: Θεσσαλονίκη, : Προτεραιότητα 1, Κόστος: 2.20€

Φάκελος2 : Προορισμός: Θεσσαλονίκη, : Προτεραιότητα 3, Κόστος: 2.60€

ΟγκώδεςΑντικείμενο1 : Προορισμός: Αθήνα, : Διαστάσεις 0.7 x 0.5 x 1.0m, Κόστος: 8.00€

ΟγκώδεςΑντικείμενο2 : Προορισμός: Θεσσαλονίκη, : Διαστάσεις 1.0 x 0.5 x 1.0m, Κόστος: 12.00€

ΟγκώδεςΑντικείμενο3 : Προορισμός: Αθήνα, : Διαστάσεις 2.0 x 0.7 x 0.7m, Κόστος: 20.60€

ΣΥΝΟΛΙΚΟ ΚΟΣΤΟΣ: 82.65€

Υπόδειξη: Να χρησιμοποιηθεί το αρχείο `4_code_template.py` ως οδηγός επίλυσης.

Γενικές Υποδείξεις:

I) Για τις απαντήσεις της εργασίας μπορείτε να ανατρέξετε στη συμπληρωματική βιβλιογραφία που δίνεται και στα βοηθητικά κείμενα που υπάρχουν στον δικτυακό τόπο / portal της θεματικής ενότητας. Συνιστάται να προσθέσετε στο τέλος της εργασίας σας κατάλογο βιβλιογραφίας.

II) Οδηγίες σχετικές με τον κώδικα

- Το όνομα κάθε .py αρχείου να περιλαμβάνει το επώνυμό σας με λατινικούς χαρακτήρες, το χαρακτήρα της υπογράμμισης και τον αριθμό του συγκεκριμένου υποερωτήματος (π.χ. αν το επώνυμό σας είναι Γεωργίου, τότε ο κώδικας για την υποεργασία 1β θα έχει το όνομα Georgiou_1b.py). Κάθε αρχείο κώδικα που θα παραδοθεί θα πρέπει τουλάχιστον να περνάει τη φάση της διερμηνείας χωρίς συντακτικά σφάλματα.
- Τα αρχεία .py θα πρέπει να τα ανοίξετε και να τα επεξεργαστείτε με το ολοκληρωμένο περιβάλλον ανάπτυξης κώδικα IDLE. Ο κώδικας να είναι επαρκώς σχολιασμένος, σωστά στοιχισμένος και ενσωματωμένος μέσα στο έγγραφο Word, με τις απαντήσεις σας σε γραμματοσειρά courier.
- Στο έγγραφο της απάντησής σας και στο αρχείο του κώδικα θα πρέπει να δίνεται ολόκληρο το πρόγραμμα, επισημαίνοντας με σχόλια πού απαντάτε κάθε ερώτημα ώστε να θεωρούνται πλήρεις οι απαντήσεις.
- Όλα τα .py αρχεία με τον πηγαίο κώδικα και το .doc αρχείο κειμένου να υποβληθούν μέσω της πλατφόρμας <https://study.eap.gr>.

III) Τρόπος παράδοσης εργασίας:

α) Οι απαντήσεις πρέπει να είναι γραμμένες με χρήση **επεξεργαστή κειμένου** (π.χ. **Word**) σε σελίδες **διαστάσεων A4 χωρίς χρώματα**. Το αρχείο να περιέχει ως **πρώτη σελίδα** το κείμενο του **Εντύπου Υποβολής – Αξιολόγησης** και ως δεύτερη σελίδα τον τίτλο «Σχόλια προς τον φοιτητή» (θα συμπληρωθεί από τον καθηγητή σας). Οι απαντήσεις στις υποεργασίες θα αρχίζουν από την τρίτη σελίδα, **χωρίς να επαναλαμβάνονται οι εκφωνήσεις**. Κάθε υποεργασία θα αρχίζει από νέα σελίδα. Για την απάντησή σας θα πρέπει να χρησιμοποιείτε υποχρεωτικά το **Πρότυπο Υποβολής Γραπτής Εργασίας**.

β) Το .doc αρχείο κειμένου να υποβληθεί στη διεύθυνση <https://study.eap.gr> με **όνομα αρχείου το επώνυμό σας με λατινικούς χαρακτήρες και τον Αριθμό Μητρώου σας**, π.χ. Ioannou_82345.

IV) Η καλή παρουσίαση της εργασίας λαμβάνεται υπόψη στην αξιολόγηση της εργασίας.
