

**Ε.Α.Π./ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ**

**1η ΓΡΑΠΤΗ ΕΡΓΑΣΙΑ**

**ΥΠΟΔΕΙΓΜΑΤΙΚΗ ΛΥΣΗ**

**ΠΑΠΑΘΕΟΔΩΡΟΥ ΚΩΝΣΤΑΝΤΙΝΟΣ**

**[k.papathodorou@arnos.gr](mailto:k.papathodorou@arnos.gr)**

**ΑΚΑΔΗΜΑΪΚΟΥ ΕΤΟΥΣ 2021-2022**

**11/10/2021**

**Ημερομηνία παράδοσης εργασίας: Κυριακή 7/11/2021**

**Καταληκτική ημερομηνία παραλαβής: Τετάρτη <sup>1</sup> 10/11/2021**

**Ημερομηνία ανάρτησης ενδεικτικών λύσεων: Σάββατο 13/11/2021**

**Καταληκτική ημερομηνία αποστολής σχολίων στον φοιτητή: Κυριακή 28/11/2021**

---

<sup>1</sup> 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.

*Η Γνώση με τρόπο απλό και κατανοητό!*

**ΥΠΟΕΡΓΑΣΙΑ 1.**

(βαθμοί 25)

Εισαγωγή στην Python, Δομές Επανάληψης

**ΥΠΟΕΡΓΑΣΙΑ 2.**

(βαθμοί 25)

Είσοδος δεδομένων, Έλεγχος ροής εκτέλεσης

**ΥΠΟΕΡΓΑΣΙΑ 3.**

(βαθμοί 25)

Διαχείριση Συμβολοσειρών

**ΥΠΟΕΡΓΑΣΙΑ 4.**

(βαθμοί 25)

Είσοδος δεδομένων, Έλεγχος ροής εκτέλεσης

**ΣΥΝΟΛΟ**

**(βαθμοί 100)**

*Η Γνώση με τρόπο απλό και κατανοητό!*

**ΥΠΟΕΡΓΑΣΙΑ 1.**

(βαθμοί 25)

Να γράψετε πρόγραμμα, το οποίο να υπολογίζει και να εκτυπώνει το άθροισμα των ακόλουθων εννέα ακεραίων αριθμών:  $9 + 99 + 999 + \dots + 999999999$ .

Υπόδειξη: Να υλοποιηθεί με δομή επανάληψης.

**ΑΠΑΝΤΗΣΗ**

```
# ΥΠΟΕΡΓΑΣΙΑ 1
# Να γράψετε πρόγραμμα, το οποίο να υπολογίζει και να εκτυπώνει το
# άθροισμα των ακόλουθων
# εννέα ακεραίων αριθμών: 9 + 99 + 999 + ... + 999999999.
# Υπόδειξη: Να υλοποιηθεί με δομή επανάληψης.

#Αρχικοποιώ το σύνολο και τον αριθμό 9
sum=0
n=9

#Συνάρτηση range(). Επιτρέπει τη δημιουργία επαναλήψιμων ακολουθιών
#ακέραιων αριθμών.
#Μεταβλητή i λαμβάνει τιμές από 0 έως και 8
for i in range(9):
    sum = sum + n
    n=(n*10)+9

#Εκτυπώνω το άθροισμα
print("Το άθροισμα των ακόλουθων εννέα ακεραίων αριθμών: 9 + 99 + 999
+ ... + 999999999 είναι = ", sum)
```

Η Γνώση με τρόπο απλό και κατανοητό!

**ΥΠΟΕΡΓΑΣΙΑ 2.****(βαθμοί 25)**

Να γραφεί πρόγραμμα, το οποίο να διαβάζει έναν θετικό ακέραιο αριθμό και να εμφανίζει το άθροισμα των ψηφίων του αριθμού. Για παράδειγμα, ο αριθμός 462459 έχει άθροισμα ψηφίων:  
 $4 + 6 + 2 + 4 + 5 + 9 = 30$ .

Υπόδειξη: Να ελέγξετε μέσω κατάλληλου ελέγχου (αμυντικός προγραμματισμός) την είσοδο του χρήστη και το πρόγραμμα να προχωρήσει σε υπολογισμό μόνο στην περίπτωση που εισαχθεί θετικός ακέραιος.

**ΑΠΑΝΤΗΣΗ**

```
# ΥΠΟΕΡΓΑΣΙΑ 2. (βαθμοί 25)
# Να γραφεί πρόγραμμα, το οποίο να διαβάζει έναν θετικό ακέραιο αριθμό
και να εμφανίζει το
# άθροισμα των ψηφίων του αριθμού. Για παράδειγμα, ο αριθμός 462459
έχει
# άθροισμα ψηφίων: 4 + 6 + 2 + 4 + 5 + 9 = 30.
# Υπόδειξη: Να ελέγξετε μέσω κατάλληλου ελέγχου (αμυντικός
προγραμματισμός) την είσοδο
# του χρήστη και το πρόγραμμα να προχωρήσει σε υπολογισμό μόνο στην
περίπτωση που εισαχθεί θετικός ακέραιος.

# Αμυντικός προγραμματισμός
while True:

    # Εισαγωγή Αριθμού
    user_input = input("Παρακαλώ εισάγετε έναν θετικό ακέραιο
αριθμό:")

    # isdigit(). Ελέγχει αν όλοι οι χαρακτήρες είναι ψηφία:
    if user_input.isdigit():
        break # Βγαίνει από το while

# Μετατρέπω τον αριθμό σε String
num_str = str(user_input)
sum = 0

# Διαπερνάω τον αριθμό
for i in range(0, len(num_str)):
    # Υπολογίζω το άθροισμα μετατρέποντας το String σε Integer
    sum = sum + int(num_str[i])

# Εμφανίζω το άθροισμα των ψηφίων του αριθμού
print("Το άθροισμα των ψηφίων του αριθμού " + user_input + " είναι: "
+ str(sum))
```

**Η Γνώση με τρόπο απλό και κατανοητό!**

**ΥΠΟΕΡΓΑΣΙΑ 3.**

(βαθμοί 25)

α) Να γραφεί πρόγραμμα, το οποίο να διαβάζει μία συμβολοσειρά ως είσοδο από τον χρήστη. Να εφαρμοστεί αμυντικός προγραμματισμός ώστε η συμβολοσειρά να μην είναι κενή.

β) Στη συνέχεια, να τυπώνει τη συμβολοσειρά αφού έχει αφαιρέσει τον χαρακτήρα στη θέση  $N$  (1 έως και το μήκος της συμβολοσειράς) που θα υποδείξει ο χρήστης. Να εφαρμοστεί αμυντικός προγραμματισμός ώστε η θέση που υποδεικνύει ο χρήστης να είναι έγκυρη.

γ) Τέλος, να ζητά από τον χρήστη έναν χαρακτήρα και να επιστρέφει το πλήθος και το ποσοστό των εμφανίσεων του συγκεκριμένου χαρακτήρα στην αρχική συμβολοσειρά.

**Παράδειγμα εκτέλεσης:**

```
Παρακαλώ εισάγετε μία συμβολοσειρά: Παρακαταθήκη
Παρακαλώ εισάγετε τη θέση του προς αφαίρεση χαρακτήρα: 3
Η νέα συμβολοσειρά: Παακαταθήκη
Παρακαλώ εισάγετε έναν χαρακτήρα: α
Ο χαρακτήρας α εμφανίζεται 4 φορές στη συμβολοσειρά: Παρακαταθήκη
Το ποσοστό των εμφανίσεων του χαρακτήρα α στην αρχική συμβολοσειρά είναι 33%

Ολοκληρώθηκε η εκτέλεση του προγράμματος
```

**ΑΠΑΝΤΗΣΗ**

```
# ΥΠΟΕΡΓΑΣΙΑ 3. (βαθμοί 25)
# α) Να γραφεί πρόγραμμα, το οποίο να διαβάζει μία συμβολοσειρά ως είσοδο από
τον χρήστη.
# Να εφαρμοστεί αμυντικός προγραμματισμός ώστε η συμβολοσειρά να μην είναι
κενή.

# β) Στη συνέχεια, να τυπώνει τη συμβολοσειρά αφού έχει αφαιρέσει τον
χαρακτήρα στη θέση N
# (1 έως και το μήκος της συμβολοσειράς) που θα υποδείξει ο χρήστης. Να
εφαρμοστεί αμυντικός
# προγραμματισμός ώστε η θέση που υποδεικνύει ο χρήστης να είναι έγκυρη.

# γ) Τέλος, να ζητά από τον χρήστη έναν χαρακτήρα και να επιστρέφει το πλήθος
και το ποσοστό
# των εμφανίσεων του συγκεκριμένου χαρακτήρα στην αρχική συμβολοσειρά.

while True:

    # διαβάζει μία συμβολοσειρά ως είσοδο από τον χρήστη
    user_input = input("Παρακαλώ εισάγετε μία συμβολοσειρά:")

    # Να εφαρμοστεί αμυντικός προγραμματισμός ώστε η συμβολοσειρά να μην
είναι κενή
    if user_input:
        break # Βγαίνει από το while
```

*Η Γνώση με τρόπο απλό και κατανοητό!*

```
while True:
    try:
        char_input = int(input("Παρακαλώ εισάγετε τη θέση του προς αφαίρεση
        χαρακτήρα:"))
        # Να εφαρμοστεί αμυντικός προγραμματισμός ώστε η θέση που υποδεικνύει
        ο χρήστης να είναι έγκυρη.
        if char_input >= 0 and char_input < len(user_input):
            break
        else:
            print("Δεν εισαγάγετε σωστή θέση!")
    except:
        print("Δεν εισαγάγετε ακέραιο!")
# Χρησιμοποιούμε τεμαχισμό, ξαναχτίζοντας τη συμβολοσειρά μείον τον χαρακτήρα
newstr = user_input[:char_input - 1] + user_input[char_input:]
print("Η νέα συμβολοσειρά:" + newstr)

# Τέλος, να ζητά από τον χρήστη έναν χαρακτήρα
while True:
    try:
        char = input("Παρακαλώ εισάγετε έναν χαρακτήρα:")
        if char.isalpha():
            break
    except:
        print("Δεν εισαγάγετε χαρακτήρα!")

# Να επιστρέφει το πλήθος των εμφανίσεων του συγκεκριμένου χαρακτήρα στην
αρχική συμβολοσειρά.
times = user_input.count(char)
print("Ο χαρακτήρας " + char + " εμφανίζεται " + str(times) + " φορές στη
συμβολοσειρά: " + user_input)

# Να επιστρέφει το ποσοστό των εμφανίσεων του συγκεκριμένου χαρακτήρα στην
αρχική συμβολοσειρά.
percentage = 100 * times / len(user_input)
percentage = int(percentage)
print("Το ποσοστό των εμφανίσεων του χαρακτήρα " + char + " στην αρχική
συμβολοσειρά είναι " + str(percentage) + "%")
```

*Η Γνώση με τρόπο απλό και κατανοητό!*

**ΥΠΟΕΡΓΑΣΙΑ 4.**

(βαθμοί 25)

Υποθέτουμε ότι μία μηχανή προσφέρει τέσσερα διαφορετικά είδη ροφημάτων (καφέ, καφέ με γάλα, σοκολάτα και σοκολάτα με γάλα), που κοστίζουν 1.50€, 1.80€, 2.10€ και 2.40€ αντίστοιχα. Η μηχανή δέχεται κέρματα των 10, 20 και 50 λεπτών, του ενός (1) ευρώ και των δύο (2) ευρώ, καθώς και χαρτονομίσματα των 5€ και 10€, και επιστρέφει ρέστα χρησιμοποιώντας μόνο κέρματα. Να υλοποιηθεί πρόγραμμα, το οποίο:

α) Να εμφανίζει κατάλογο επιλογής (μενού) των προσφερόμενων ειδών (αριθμούμενα από το 1 έως το 4) με το αντίστοιχο αντίτιμο για το καθένα, την επιλογή 0 για έξοδο από το πρόγραμμα και στη συνέχεια διαβάζει την επιλογή του χρήστη (είδος που προτιμά ή έξοδος), εφαρμόζοντας αμυντικό προγραμματισμό προκειμένου να διασφαλιστεί ότι ο χρήστης εισάγει τιμή μεταξύ του 0 και του 4.

β) Στη συνέχεια, να εμφανίζει στην οθόνη το ποσό που απαιτείται για την πληρωμή του είδους που επέλεξε ο χρήστης. Ακολουθώντας, κατά την εισαγωγή του ποσού από τον χρήστη να χρησιμοποιηθεί αμυντικός προγραμματισμός, ώστε το ποσό που θα εισαχθεί να αντιστοιχεί σε αποδεκτό κέρμα ή χαρτονόμισμα. Το πρόγραμμα να ελέγχει εάν το ποσό που εισήχθη είναι μεγαλύτερο ή ίσο του απαιτούμενου ποσού. Στην περίπτωση που έχει εισαχθεί ποσό μικρότερο από το απαιτούμενο, το πρόγραμμα τυπώνει κατάλληλο μήνυμα (με το επιπλέον ποσό που πρέπει να εισαχθεί) και προτρέπει τον χρήστη να εισάγει περισσότερα χρήματα. Αυτή η διαδικασία συνεχίζεται μέχρι να εισαχθεί συνολικά ποσό ίσο με ή μεγαλύτερο από το ποσό που απαιτείται για την αγορά του είδους που επελέγη από το χρήστη.

γ) Να υπολογίζει το υπόλοιπο ποσό (ρέστα) που πρέπει να επιστραφεί (διαφορά του συνολικού εισαχθέντος ποσού από το αντίτιμο) και να τυπώνει κατάλληλο μήνυμα που ενημερώνει το χρήστη για το ποσό που θα του επιστραφεί.

δ) Να υπολογίζει το ελάχιστο πλήθος κερμάτων που θα επιστραφούν στον χρήστη ως υπόλοιπο (ρέστα) και να τυπώνει μήνυμα με πόσα και ποιας αξίας κέρματα πραγματοποιείται αυτό. Το πρόγραμμα θα δέχεται από τον χρήστη τα ποσά σε ευρώ με ακρίβεια δυο δεκαδικών ψηφίων, αλλά θα χειρίζεται τα ποσά σε λεπτά.

Παράδειγμα εκτέλεσης:

*Η Γνώση με τρόπο απλό και κατανοητό!*

Δίνονται οι παρακάτω επιλογές:

1. Καφές: 1.5 ευρώ
2. Καφές με γάλα: 1.8 ευρώ
3. Σοκολάτα: 2.1 ευρώ
4. Σοκολάτα με γάλα: 2.4 ευρώ
0. Έξοδος

Παρακαλώ εισάγετε την επιλογή σας (1-4) ή πατήστε 0 για έξοδο: 4  
 Πρέπει να εισάγετε 2.4 ευρώ συνολικά  
 Πόσα εισάγετε; 1  
 Πρέπει να εισάγετε 1.4 ευρώ συνολικά  
 Πόσα εισάγετε; 0.8  
 ΣΦΑΛΜΑ: εισαγωγή μη έγκυρου ποσού.  
 Παρακαλώ, εισάγετε μία έγκυρη τιμή: 0.1 / 0.2 / 0.5 / 1 / 2 / 5 / 10  
 Πρέπει να εισάγετε 1.4 ευρώ συνολικά  
 Πόσα εισάγετε; 5  
 Επιστροφή 3.6 ευρώ  
 Παρακαλώ πάρτε  
 δίευρα : 1  
 μονόευρα : 1  
 πενήντάλεπτα: 1  
 δεκάλεπτα: 1

$$5 + 1 - 2.4 = 3.6 \text{ ευρώ ρέστα}$$

Ολοκληρώθηκε η εκτέλεση του προγράμματος

## ΑΠΑΝΤΗΣΗ

```
# ΥΠΟΕΡΓΑΣΙΑ 4. (βαθμοί 25)
# Υποθέτουμε ότι μία μηχανή προσφέρει τέσσερα διαφορετικά είδη ροφημάτων
# (καφέ, καφέ με
# γάλα, σοκολάτα και σοκολάτα με γάλα), που κοστίζουν 1.50€, 1.80€, 2.10€ και
# 2.40€
# αντίστοιχα. Η μηχανή δέχεται κέρματα των 10, 20 και 50 λεπτών, του ενός (1)
# ευρώ και των δύο
# (2) ευρώ, καθώς και χαρτονομίσματα των 5€ και 10€, και επιστρέφει ρέστα
# χρησιμοποιώντας
# μόνο κέρματα. Να υλοποιηθεί πρόγραμμα, το οποίο:

# α) Να εμφανίζει κατάλογο επιλογής (μενού) των προσφερόμενων ειδών
# (αριθμούμενα από το 1
# έως το 4) με το αντίστοιχο αντίτιμο για το καθένα, την επιλογή 0 για έξοδο
# από το πρόγραμμα
# και στη συνέχεια διαβάζει την επιλογή του χρήστη (είδος που προτιμά ή
# έξοδος), εφαρμόζοντας
# αυντικό προγραμματισμό προκειμένου να διασφαλιστεί ότι ο χρήστης εισάγει
# τιμή μεταξύ του
# 0 και του 4.
```

```
while True:
```

```
    print("Δίνονται οι παρακάτω επιλογές:")
    print("1. Καφές: 1.5 ευρώ")
    print("2. Καφές με γάλα: 1.8 ευρώ")
    print("3. Σοκολάτα: 2.1 ευρώ")
    print("4. Σοκολάτα με γάλα: 2.4 ευρώ")
    print("0. Έξοδος")

    try:
        user_input = int(input("Παρακαλώ εισάγετε την επιλογή σας (1-4) ή
πατήστε 0 για έξοδο:"))

        if user_input >= 0 and user_input <= 4:
            break
        else:
            print("Δεν εισαγάγετε σωστό αριθμό!")
```

**Η Γνώση με τρόπο απλό και κατανοητό!**



```
        print("")
    except:
        print("Δεν εισαγάγετε αριθμό!")
        print("")

# β) Στη συνέχεια, να εμφανίζει στην οθόνη το ποσό που απαιτείται για την
# πληρωμή του είδους
# που επέλεξε ο χρήστης. Ακολουθως, κατά την εισαγωγή του ποσού από τον
# χρήστη να
# χρησιμοποιηθεί αμυντικός προγραμματισμός, ώστε το ποσό που θα εισαχθεί να
# αντιστοιχεί σε
# αποδεκτό κέρμα ή χαρτονόμισμα. Το πρόγραμμα να ελέγχει εάν το ποσό που
# εισήχθη είναι
# μεγαλύτερο ή ίσο του απαιτούμενου ποσού. Στην περίπτωση που έχει εισαχθεί
# ποσό μικρότερο
# από το απαιτούμενο, το πρόγραμμα τυπώνει κατάλληλο μήνυμα (με το επιπλέον
# ποσό που
# πρέπει να εισαχθεί) και προτρέπει τον χρήστη να εισάγει περισσότερα
# χρήματα. Αυτή η
# διαδικασία συνεχίζεται μέχρι να εισαχθεί συνολικά ποσό ίσο με ή μεγαλύτερο
# από το ποσό που
# απαιτείται για την αγορά του είδους που επελέγη από το χρήστη.
price = 0
if user_input == 1:
    price = 1.5
elif user_input == 2:
    price = 1.8
elif user_input == 3:
    price = 2.1
elif user_input == 4:
    price = 2.4
else:
    exit()
print("Πρέπει να εισάγετε " + str(price) + " ευρώ συνολικά")

# κατά την εισαγωγή του ποσού από τον χρήστη να χρησιμοποιηθεί αμυντικός
# προγραμματισμός,
# ώστε το ποσό που θα εισαχθεί να αντιστοιχεί σε αποδεκτό κέρμα ή
# χαρτονόμισμα.
while True:

    try:
        money_input = float(input("Πόσα εισάγετε;"))

        if money_input == 0.1 or money_input == 0.2 or money_input == 0.5 or
money_input == 1 or money_input == 2 \
            or money_input == 5 or money_input == 10:
            # Το πρόγραμμα να ελέγχει εάν το ποσό που εισήχθη είναι
            # μεγαλύτερο ή ίσο του απαιτούμενου ποσού
            if money_input < price:
                # Στην περίπτωση που έχει εισαχθεί ποσό μικρότερο από το
                # απαιτούμενο, το πρόγραμμα τυπώνει μήνυμα
                # (με το επιπλέον ποσό που πρέπει να εισαχθεί) και προτρέπει
                # τον χρήστη να εισάγει περισσότερα χρήματα.
                price = round(price - money_input, 1)
                print("Πρέπει να εισάγετε " + str(price) + " ευρώ συνολικά")
            else:
                break
        else:
            print("      ΣΦΑΛΜΑ: εισαγωγή μη έγκυρου ποσού.")
            print("      Παρακαλώ, εισάγετε μία έγκυρη τιμή: 0.1 / 0.2 / 0.5 /
```

## Η Γνώση με τρόπο απλό και κατανοητό!

```
1 / 2 / 5 / 10")
    print("Πρέπει να εισάγετε " + str(price) + " ευρώ συνολικά")
except:
    print("      ΣΦΑΛΜΑ: εισαγωγή μη έγκυρου ποσού.")
    print("      Παρακαλώ, εισάγετε μία έγκυρη τιμή: 0.1 / 0.2 / 0.5 / 1 /
2 / 5 / 10")
    print("Πρέπει να εισάγετε " + str(price) + " ευρώ συνολικά")

# γ) Να υπολογίζει το υπόλοιπο ποσό (ρέστα) που πρέπει να επιστραφεί (διαφορά
του συνολικού
# εισαχθέντος ποσού από το αντίτιμο) και να τυπώνει κατάλληλο μήνυμα που
ενημερώνει το
# χρήστη για το ποσό που θα του επιστραφεί.
resta = round(money_input - price,1)
print("Επιστροφή " + str(resta) + " ευρώ")

# δ) Να υπολογίζει το ελάχιστο πλήθος κερμάτων που θα επιστραφούν στον χρήστη
ως υπόλοιπο
# (ρέστα) και να τυπώνει μήνυμα με πόσα και ποιας αξίας κέρματα
πραγματοποιείται αυτό. Το
# πρόγραμμα θα δέχεται από τον χρήστη τα ποσά σε ευρώ με ακρίβεια δυο
δεκαδικών ψηφίων,
# αλλά θα χειρίζεται τα ποσά σε λεπτά.
pente = 0
dyo = 0
ena = 0
penynta = 0
eikosi = 0
deka = 0

while resta > 0:
    if resta >= 5:
        resta = round(resta - 5, 1)
        pente = pente + 1
    elif resta >= 2:
        resta = round(resta - 2, 1)
        dyo = dyo + 1
    elif resta >= 1:
        resta = round(resta - 1,1)
        ena = ena + 1
    elif resta >= 0.5:
        resta = round(resta - 0.5,1)
        penynta = penynta + 1
    elif resta >= 0.2:
        resta = round(resta - 0.2,1)
        eikosi = eikosi + 1
    elif resta >= 0.1:
        resta = round(resta - 0.1,1)
        deka = deka + 1
    else:
        break

print("Παρακαλώ πάρτε")
if pente > 0: print(" πεντάευρα : " + str(pente))
if dyo > 0: print(" δίευρα : " + str(dyo))
if ena > 0: print(" μονόευρα : " + str(ena))
if penynta > 0: print(" πενηντάλεπτα : " + str(penynta))
if eikosi > 0: print(" εικοσάλεπτα : " + str(eikosi))
if deka > 0: print(" δεκάλεπτα : " + str(deka))
```

**Η Γνώση με τρόπο απλό και κατανοητό!**

## 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.

\*\*\*\*\*