

Ασκήσεις στην Python

1. Γράψτε κώδικα ο οποίος να περιέχει έναν βρόχο for για εκτύπωση των αριθμών από το 1 έως και το 10.

```
for i in range(1, 11):  
    print (i)
```

2. Γράψτε κώδικα ο οποίος να περιέχει έναν βρόχο for για αντίστροφη εκτύπωση των προηγούμενων αριθμών (από το 10 έως και το 1).

```
for i in range(10, 0, -1):  
    print (i)
```

3. Γράψτε κώδικα ο οποίος να διαβάζει την ηλικία σας και στη συνέχεια να τυπώνει τους άρτιους αριθμούς μέχρι να φτάσει την ηλικία σας.

```
# Διάβασε την αρχική ηλικία  
initial_age = int(input("Δώσε την αρχική σου ηλικία: "))  
  
# Αρχικοποίηση μετρητή  
counter = 0  
  
# Βρόχος while  
while counter <= initial_age:  
    # Έλεγχος για άρτιο αριθμό  
    if counter % 2 == 0:  
        print(counter)  
  
    # Αύξηση του μετρητή  
    counter += 1
```

4. Να γράψετε πρόγραμμα το οποίο θα εκτυπώνει όλα τα πολλαπλάσια του 8 από το 0 μέχρι και το 50.

```
# Χρησιμοποιούμε έναν βρόχο for για να επαναλάβουμε τις τιμές από 0 έως 50
for i in range(0, 51):
    # Έλεγχος αν ο αριθμός είναι πολλαπλάσιο του 8
    if i % 8 == 0:
        print(i)
```

5. Να γράψετε πρόγραμμα το οποίο θα διαβάζει έναν ακέραιο αριθμό και θα εκτυπώνει την προπαίδεια αυτού του αριθμού.

```
# Διάβασε έναν ακέραιο αριθμό από τον χρήστη
num = int(input("Δώσε έναν ακέραιο αριθμό: "))
# Αρχικοποίηση της προπαίδειας σε 1
factorial = 1
# Υπολογισμός της προπαίδειας
for i in range(1, num + 1):
    factorial *= i
# Εκτύπωση της προπαίδειας
print(f"Η προπαίδεια του {num} είναι: {factorial}")
```

6. Να γράψετε πρόγραμμα το οποίο θα εκτυπώνει όλους τους άρτιους από το 50 μέχρι το -10.

```
# Χρησιμοποιούμε έναν βρόχο for με βήμα -2 για να επαναλάβουμε τις τιμές από
το 50 μέχρι το -10

for i in range(50, -11, -2):

    print(i)
```

7. Να γράψετε πρόγραμμα το οποίο θα διαβάζει 10 ακεραίους από το πληκτρολόγιο και θα υπολογίζει και θα εκτυπώνει το μέσο όρο μόνο των αρτίων αριθμών που διαβάστηκαν.

```
# Αρχικοποίηση του αθροίσματος και του μετρητή των αρτίων

sum_even = 0

count_even = 0

# Διάβασε 10 ακέραιους αριθμούς από το χρήστη

for _ in range(10):

    number = int(input("Δώσε έναν ακέραιο αριθμό: "))

    # Έλεγχε αν είναι άρτιος

    if number % 2 == 0:

        sum_even += number

        count_even += 1

# Υπολόγισε το μέσο όρο αρτίων αριθμών (αν υπάρχουν αρτιοί)

if count_even > 0:

    average_even = sum_even / count_even

    print(f"Ο μέσος όρος των αρτίων αριθμών είναι: {average_even}")

else:

    print("Δεν δώσατε κανέναν άρτιο αριθμό.")
```

8. Να γράψετε πρόγραμμα το οποίο διαβάζει 50 τυχαίους ακέριους αριθμούς και θα εκτυπώνει τα ποσοστά των μονοψήφιων, διψήφιων και τριψήφιων αριθμών που διαβάστηκαν.

```
import random

# Αρχικοποίηση μετρητών

count_single_digit = 0

count_double_digit = 0

count_triple_digit = 0

# Διάβασε 50 τυχαίους αριθμούς

for _ in range(50):

    number = random.randint(1, 999)

    print(f"Τυχαίος αριθμός: {number}")

    # Καταμέτρηση ανάλογα με τον αριθμό των ψηφίων

    num_digits = len(str(number))

    if num_digits == 1:

        count_single_digit += 1

    elif num_digits == 2:

        count_double_digit += 1

    elif num_digits == 3:

        count_triple_digit += 1

# Υπολογισμός ποσοστών

total_numbers = count_single_digit + count_double_digit + count_triple_digit

percentage_single_digit = (count_single_digit / total_numbers) * 100

percentage_double_digit = (count_double_digit / total_numbers) * 100

percentage_triple_digit = (count_triple_digit / total_numbers) * 100
```

```
# Εκτύπωση αποτελεσμάτων
print(f"\nΠοσοστό μονοψήφιων: {percentage_single_digit}%")
print(f"Ποσοστό διψήφιων: {percentage_double_digit}%")
print(f"Ποσοστό τριψήφιων: {percentage_triple_digit}%")
```

9. Να γράψετε πρόγραμμα το οποίο εκτυπώνει όλους τους θετικούς διψήφιους ακεραίους. Τροποποιήστε το πρόγραμμα ώστε να υπολογίζει και να εμφανίζει το άθροισμα των αριθμών αυτών καθώς και τον μέσο όρο τους.

```
# Αρχικοποίηση μετρητών
sum_of_digits = 0
count_of_numbers = 0

# Εκτύπωση διψήφιων αριθμών και υπολογισμός άθροισματος και μέσου όρου
print("Διψήφιοι αριθμοί:")
for i in range(10, 100):
    # Έλεγχος αν είναι διψήφιος
    if i >= 10 and i <= 99:
        print(i)

# Υπολογισμός άθροισματος ψηφίων
    sum_of_digits += i % 10 + i // 10
    count_of_numbers += 1

# Υπολογισμός μέσου όρου ψηφίων
average_of_digits = sum_of_digits / count_of_numbers if count_of_numbers > 0 else 0

# Εκτύπωση αποτελεσμάτων
print(f"\nΆθροισμα ψηφίων: {sum_of_digits}")
print(f"Μέσος όρος ψηφίων: {average_of_digits}")
```

10. Να γράψετε πρόγραμμα το οποίο θα διαβάζει 20 ακεραίους από τον πληκτρολόγιο και θα υπολογίζει και θα εκτυπώνει το πλήθος των άρτιων αριθμών που διαβάστηκαν καθώς και το πλήθος των αριθμών που είναι μεγαλύτεροι από 15.

```
# Αρχικοποίηση μετρητών
count_even_greater_than_15 = 0
count_numbers_greater_than_15 = 0

# Διάβασε 20 ακέραιους αριθμούς από το χρήστη
for _ in range(20):
    number = int(input("Δώσε έναν ακέραιο αριθμό: "))

# Έλεγχος αν είναι άρτιος
if number % 2 == 0:
    count_even_greater_than_15 += 1

# Έλεγχος αν είναι μεγαλύτερος από 15
if number > 15:
    count_numbers_greater_than_15 += 1

# Εκτύπωση αποτελεσμάτων
print(f"Πλήθος άρτιων αριθμών: {count_even_greater_than_15}")
print(f"Πλήθος αριθμών μεγαλύτερων από 15: {count_numbers_greater_than_15}")
```

11. Να γράψετε πρόγραμμα που θα εκτυπώνει τους τριψήφιους αριθμούς που είναι πολλαπλάσια του 7 καθώς και το πόσοι είναι οι αριθμοί αυτοί.

```
# Αρχικοποίηση μετρητών
count_even_greater_than_15 = 0
count_numbers_greater_than_15 = 0

# Διάβασε 20 ακέραιους αριθμούς από το χρήστη
for _ in range(20):
    number = int(input("Δώσε έναν ακέραιο αριθμό: "))

# Έλεγχος αν είναι άρτιος
    if number % 2 == 0:
        count_even_greater_than_15 += 1

# Έλεγχος αν είναι μεγαλύτερος από 15
    if number > 15:
        count_numbers_greater_than_15 += 1

# Εκτύπωση αποτελεσμάτων
print(f"Πλήθος άρτιων αριθμών: {count_even_greater_than_15}")
print(f"Πλήθος αριθμών μεγαλύτερων από 15:
{count_numbers_greater_than_15}")
```

12. Να γράψετε πρόγραμμα που θα εκτυπώνει το άθροισμα των τετραγώνων των μονοψήφιων άρτιων αριθμών.

```
# Αρχικοποίηση μετρητών
count_even_greater_than_15 = 0
count_numbers_greater_than_15 = 0
# Διάβασε 20 ακέραιους αριθμούς από το χρήστη
for _ in range(20):
    number = int(input("Δώσε έναν ακέραιο αριθμό: "))
    # Έλεγχος αν είναι άρτιος
    if number % 2 == 0:
        count_even_greater_than_15 += 1
    # Έλεγχος αν είναι μεγαλύτερος από 15
    if number > 15:
        count_numbers_greater_than_15 += 1
# Εκτύπωση αποτελεσμάτων
print(f"Πλήθος άρτιων αριθμών: {count_even_greater_than_15}")
print(f"Πλήθος αριθμών μεγαλύτερων από 15:
{count_numbers_greater_than_15}")
```


13. Σε ένα αγώνα ρίψης ακοντίου, διεξάγεται ο προκριματικός γύρος με τη συμμετοχή 14 αθλητών. Στην τελική φάση προκρίνονται όσοι αθλητές πετύχουν επίδοση άνω των 80 μέτρων. Να γράψετε πρόγραμμα που να διαβάζει την επίδοση κάθε αθλητή, να υπολογίζει και να εμφανίζει πόσοι αθλητές πέρασαν το όριο.

```
# Διάβασε την επίδοση κάθε αθλητή
for i in range(1, 15):
    performance = float(input(f"Δώσε την επίδοση του {i}ου αθλητή σε μέτρα: "))
    # Έλεγχος αν πέρασε το όριο
    if performance > 80:
        qualified_count += 1
# Εκτύπωση αποτελέσματος
print(f"\n Πέρασαν το όριο των 80 μέτρων {qualified_count} αθλητές.")
```

1. Να γράψετε πρόγραμμα το οποίο θα τυπώνει τους 20 πρώτους θετικούς ακέραιους αριθμούς με χρήση πρώτα της `for` και στη συνέχεια της `while`.

```
# Εκτύπωση με χρήση for
print("Χρήση βρόχου for:")
for i in range(1, 41, 2):
    print(i)
# Εκτύπωση με χρήση while
print("\nΧρήση βρόχου while:")
counter = 1
while counter <= 40:
    print(counter)
    counter += 2
```

Σε αυτόν τον κώδικα, ο βρόχος `for` χρησιμοποιείται για να εκτυπώσει τους πρώτους 20 θετικούς ακέραιους αριθμούς με βήμα 2 (από 1 έως 41). Ο βρόχος `while` επίσης εκτυπώνει τους ίδιους αριθμούς, με χρήση μιας μεταβλητής `counter` που αυξάνεται κατά 2 σε κάθε επανάληψη, μέχρις ότου φτάσει στο 41.

2. Να γράψετε πρόγραμμα το οποίο θα διαβάζει αριθμούς από το πληκτρολόγιο μέχρι να δοθεί ο αριθμός 0. Το πρόγραμμα θα υπολογίζει και εκτυπώνει:

- α) Πόσοι αριθμοί διαβάστηκαν (εκτός του 0).
- β) Το πλήθος των ζυγών αριθμών.
- γ) Το άθροισμα και τον μέσο όρο των μονών αριθμών.

```
# Αρχικοποίηση μεταβλητών

count_numbers = 0

sum_of_even_numbers = 0

sum_of_odd_numbers = 0

min_number = float('inf')

max_number = float('-inf')

# Διάβασε αριθμούς μέχρι τον αριθμό 0

while True:

    number = float(input("Δώσε έναν αριθμό (0 για τερματισμό): "))

    # Έλεγχος αν ο αριθμός είναι 0

    if number == 0:

        break

    # Υπολογισμός απαιτούμενων πληροφοριών

    count_numbers += 1

    if number % 2 == 0:

        sum_of_even_numbers += number

    else:

        sum_of_odd_numbers += number

    min_number = min(min_number, number)

    max_number = max(max_number, number)

# Υπολογισμός μέσου όρου μονών αριθμών

average_of_odd_numbers = sum_of_odd_numbers / count_numbers if
count_numbers > 0 else 0

# Εκτύπωση αποτελεσμάτων

print(f"\nΠλήθος αριθμών (εκτός του 0): {count_numbers}")
```

```
print(f"Άθροισμα ηυγών αριθμών: {sum_of_even_numbers}")
print(f"Άθροισμα μονών αριθμών: {sum_of_odd_numbers}")
print(f"Ελάχιστος αριθμός: {min_number}")
print(f"Μέγιστος αριθμός: {max_number}")
print(f"Άκροισμα μονών αριθμών: {sum_of_odd_numbers}")
print(f"Μέσος όρος μονών αριθμών: {average_of_odd_numbers}")
```

3. Να γράψετε πρόγραμμα το οποίο θα διαβάζει το όνομα και τους βαθμούς ενός μαθητή της Α' Λυκείου σε όλα τα μαθήματα στο Α' Τετράμηνο και στη συνέχεια θα υπολογίζει και θα εκτυπώνει:

- α) τον μέσο όρο του τετραμήνου (15 μαθήματα).
- β) το πλήθος των μαθημάτων που είχε βαθμό πάνω από 16
- γ) τον μεγαλύτερο και τον μικρότερο βαθμό
- δ) το ποσοστό των μαθημάτων που πήρε βαθμό από 18-20.

Αρχικοποίηση μεταβλητών

```
total_students = 15
```

```
total_grades = 0
```

```
sum_of_grades_over_16 = 0
```

```
max_grade = float('-inf')
```

```
min_grade = float('inf')
```

```
count_grades_18_to_20 = 0
```

Διάβασε τα ονόματα και τους βαθμούς των μαθητών

```
for i in range(total_students):
```

```
    name = input(f"Δώσε το όνομα του {i+1}ου μαθητή: ")
```

```
    grade = float(input(f"Δώσε το βαθμό του {i+1}ου μαθητή στο Α' Τρίμηνο: "))
```

Υπολογισμός απαιτούμενων πληροφοριών

```
    total_grades += grade
```

Σε αυτόν τον κώδικα, χρησιμοποιούμε έναν βρόχο `for` για να διαβάσουμε τα ονόματα και τους βαθμούς των μαθητών. Στη συνέχεια, υπολογίζουμε τις ζητούμενες πληροφορίες και τις εκτυπώνουμε.

```
if grade > 16:

    sum_of_grades_over_16 += grade

max_grade = max(max_grade, grade)

min_grade = min(min_grade, grade)

if 18 <= grade <= 20:

    count_grades_18_to_20 += 1

# Υπολογισμός μέσου όρου

average_grade = total_grades / total_students

# Εκτύπωση αποτελεσμάτων

print(f"\nΟ μέσος όρος του τετραμήνου είναι: {average_grade:.2f}")

print(f"Το άθροισμα των βαθμών πάνω από 16 είναι:
{sum_of_grades_over_16:.2f}")

print(f"Ο μεγαλύτερος βαθμός είναι: {max_grade:.2f}")

print(f"Ο μικρότερος βαθμός είναι: {min_grade:.2f}")

print(f"Το ποσοστό των βαθμών από 18-20 είναι: {count_grades_18_to_20 /
total_students * 100:.2f}%")
```

4. Να γράψετε πρόγραμμα που να βρίσκει σε ποιον όρο το άθροισμα $1+2+3+4+\dots$ γίνεται μεγαλύτερο του 2000.

```
# Αρχικοποίηση μεταβλητών
```

```
sum_result = 0
```

```
term = 1
```

```
target_sum = 2000
```

```
# Βρόχος while για τον υπολογισμό του ορίου
```

```
while sum_result <= target_sum:
```

```
    sum_result += term
```

```
    term += 1
```

```
# Εκτύπωση του αποτελέσματος
```

```
print(f"Το άθροισμα 1+2+3+4+... γίνεται μεγαλύτερο από 2000 στον όρο: {term - 1}")
```

Σε αυτόν τον κώδικα, ο όρος `term` αυξάνεται κατά 1 σε κάθε επανάληψη του βρόχου `while`, ενώ προστίθενται οι αντίστοιχοι όροι στο `sum_result`. Ο βρόχος επαναλαμβάνεται μέχρις ότου το `sum_result` γίνει μεγαλύτερο από τον στόχο `target_sum`. Στο τέλος, εκτυπώνεται ο όρος στον οποίο το άθροισμα έγινε μεγαλύτερο από 2000.

6. Να γράψετε πρόγραμμα που:

α) θα διαβάζει επαναληπτικά το όνομα και τον βαθμό στο μάθημα του Προγραμματισμού των μαθητών μιας τάξης μέχρι να διαβάσει τον χαρακτήρα \$ ως όνομα μαθητή.

β) θα υπολογίζει και εκτυπώνει τον μέσο όρο του τμήματος

γ) Το όνομα του μαθητή με τον μεγαλύτερο βαθμό (υπάρχει μόνο ένας)

δ) Το ποσοστό των μαθητών που πήραν βαθμό κάτω από 10

Σημείωση: Υπάρχει τουλάχιστον ένας μαθητής στο τμήμα.

Αρχικοποίηση μεταβλητών

```
total_students = 0
```

```
total_grades = 0
```

```
max_grade = float('-inf')
```

```
max_grade_name = ""
```

```
count_grades_below_10 = 0
```

Σε αυτό το πρόγραμμα, χρησιμοποιούμε έναν βρόχο `while` για να διαβάζουμε τα ονόματα και τους βαθμούς των μαθητών. Στο τέλος, υπολογίζουμε τις ζητούμενες πληροφορίες και τις εκτυπώνουμε.

Διάβασε το όνομα και τον βαθμό του κάθε μαθητή

```
while True:
```

```
    name = input("Δώσε το όνομα του μαθητή (ή $ για τερματισμό): ")
```

Έλεγχος για τερματισμό

```
    if name == "$":
```

```
        break
```

```
        grade = float(input(f"Δώσε τον βαθμό του μαθητή {name} στο Μάθημα του Προγραμματισμού: "))
```

Υπολογισμός απαιτούμενων πληροφοριών

```
    total_students += 1
```

```
    total_grades += grade
```

```
    if grade > max_grade:
```

```
        max_grade = grade
```

```
        max_grade_name = name
```

```
    if grade < 10:
```

```
        count_grades_below_10 += 1
```

Υπολογισμός μέσου όρου

```
average_grade = total_grades / total_students if total_students > 0 else 0
```

Εκτύπωση αποτελεσμάτων

```
print(f"\nΟ μέσος όρος των βαθμών είναι: {average_grade:.2f}")
```

```
print(f"Το όνομα του μαθητή με τον υψηλότερο βαθμό είναι:  
{max_grade_name}")  
  
print(f"Το ποσοστό των μαθητών με βαθμό κάτω από 10 είναι:  
{(count_grades_below_10 / total_students) * 100:.2f}%")
```

7. Το παραγοντικό ενός αριθμού n δίνεται από τον τύπο $n! = 1 * 2 * 3 * \dots * n$. Να γράψετε ένα πρόγραμμα που ζητά από το πληκτρολόγιο έναν ακέραιο και υπολογίζει το παραγοντικό του. Το παραγοντικό του 0 είναι 1. Συμπληρώστε το πρόγραμμα ώστε να ζητά συνέχεια αριθμό και να υπολογίζει το παραγοντικό του, μέχρι να δώσουμε αριθμό αρνητικό.

```
def factorial(n):  
  
    if n == 0:  
        return 1  
  
    else:  
        return n * factorial(n - 1)  
  
while True:  
  
    try:  
  
        num = int(input("Δώσε έναν ακέραιο αριθμό (αρνητικό για τερματισμό): "))  
  
        if num < 0:  
            break  
  
        result = factorial(num)  
  
        print(f"Το παραγοντικό του {num} είναι {result}")  
  
    except ValueError:  
  
        print("Παρακαλώ δώστε έναν έγκυρο ακέραιο.")
```

Σε αυτό το πρόγραμμα, χρησιμοποιείται η συνάρτηση `factorial` για τον υπολογισμό του παραγοντικού. Ο βρόχος `while` συνεχίζεται μέχρις ότου δοθεί ένας αρνητικός αριθμός. Εάν ο αριθμός είναι μη έγκυρος (όχι ακέραιος), εκτυπώνεται ένα μήνυμα λάθους και ο χρήστης καλείται να δώσει έναν έγκυρο αριθμό.

8. Από τις παρατηρήσεις ενός μελισσοκόμου προκύπτει ότι μια κοινότητα μελισσών υπό κανονικές συνθήκες αναπτύσσεται με ρυθμό 3.8% ετησίως. Αν ένας μελισσοκόμος διαθέτει μελίτσια με συνολικό πληθυσμό 1200 μέλισσες, να γράψετε πρόγραμμα που υπολογίζει και εμφανίζει σε πόσα έτη θα ξεπεράσει τη χωρητικότητα των κυψελών του που είναι 2000 μέλισσες.

```
def calculate_population(years, initial_population, growth_rate):  
  
    population = initial_population  
  
    for _ in range(years):  
  
        population += population * (growth_rate / 100)  
  
    return population  
  
initial_population = 1200  
  
growth_rate = 3.8  
  
capacity = 2000  
  
years = 0  
  
while initial_population <= capacity:  
  
    initial_population = calculate_population(years, initial_population, growth_rate)  
  
    years += 1  
  
print(f"Σε {years} χρόνια, ο αριθμός των κυψελών θα ξεπεράσει τη χωρητικότητα των 2000.")
```

Στο πρόγραμμα αυτό, η συνάρτηση `calculate_population` υπολογίζει τον αριθμό των κυψελών για ένα συγκεκριμένο έτος, λαμβάνοντας υπόψη τον ρυθμό αύξησης. Ο βρόχος `while` ελέγχει αν ο αριθμός των κυψελών είναι μικρότερος ή ίσος με τη χωρητικότητα των 2000, και ενημερώνει τον αριθμό των ετών. Όταν ο αριθμός των κυψελών ξεπεράσει τη χωρητικότητα, εκτυπώνεται το αντίστοιχο μήνυμα.

9. Να συμπληρώσετε το προηγούμενο πρόγραμμα ώστε να δίνουμε από το πληκτρολόγιο τον αρχικό αριθμό των μελισσών καθώς και τον αριθμό των μελισσών που δεν πρέπει να ξεπεράσει.

```
def calculate_population(years, initial_population, growth_rate):  
  
    population = initial_population  
  
    for _ in range(years):  
  
        population += population * (growth_rate / 100)  
  
    return population  
  
initial_population = int(input("Δώσε τον αρχικό αριθμό των μελισσιών: "))  
capacity_limit = int(input("Δώσε τον αριθμό των μελισσιών που δεν πρέπει να  
ξεπεραστεί: "))  
  
growth_rate = 3.8  
  
years = 0  
  
while initial_population <= capacity_limit:  
  
    initial_population = calculate_population(years, initial_population, growth_rate)  
  
    years += 1  
  
print(f"Σε {years} χρόνια, ο αριθμός των μελισσιών θα ξεπεράσει τον περιορισμό  
των {capacity_limit}.")
```

11. Να γράψετε πρόγραμμα που εμφανίζει τους 10 πρώτους αριθμούς της ακολουθίας Fibonacci.

Στα Μαθηματικά, οι αριθμοί Fibonacci είναι οι αριθμοί της ακέραιας ακολουθίας: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

Εξ ορισμού, οι πρώτοι δύο αριθμοί Fibonacci είναι το 0 και το 1, και κάθε επόμενος αριθμός είναι το άθροισμα των δύο προηγούμενων.

```
def fibonacci(n):  
  
    fib_sequence = [0, 1]  
  
    while len(fib_sequence) < n:  
  
        fib_sequence.append(fib_sequence[-1] + fib_sequence[-2])  
  
    return fib_sequence[:n]  
  
# Εμφάνιση των πρώτων 10 αριθμών Fibonacci  
  
result = fibonacci(10)  
  
print("Οι πρώτοι 10 αριθμοί Fibonacci είναι:", result)
```

Στη συνάρτηση `fibonacci`, δημιουργούμε μια λίστα `fib_sequence` που περιέχει τους πρώτους δύο αριθμούς Fibonacci (0 και 1). Στη συνέχεια, χρησιμοποιούμε έναν βρόχο `while` για να προσθέσουμε τους επόμενους αριθμούς στη λίστα μέχρι να φτάσουμε τον αριθμό `n`. Τέλος, εμφανίζουμε τους πρώτους 10 αριθμούς Fibonacci.

Αναφορές

- Ευσταθίου Χρήστος, *Αρχές Προγραμματισμού Υπολογιστώ σε γλώσσα Python*, ΕΠΑΛ Τρικάλων, 2021