

MEM104 Γλώσσα Προγραμματισμού Ι

1ο φυλλάδιο ασκήσεων

16 Οκτωβρίου 2020

1. Γράψτε ένα πρόγραμμα Python το οποίο τυπώνει το μήνυμα `Hot` αν η θερμοκρασία `T` είναι μεγαλύτερη από 35, το μήνυμα `Cold` αν είναι μικρότερη από 10, διαφορετικά το μήνυμα `Nice`.

Απάντηση. Δεδομένης της θερμοκρασίας `T`, αρκεί να γράψουμε

```
if T > 35:
    print('Hot')
elif T < 10:
    print('Cold')
else:
    print('Nice')
```

2. Τι συμβαίνει με το παρακάτω τμήμα κώδικα; Τι νομίζετε ότι ήθελε να γράψει ο συγγραφέας του;

```
x == 1
if x = 1:
    print('x is equal to 1')
```

Απάντηση. Ο συγγραφέας του παραπάνω κώδικα δεν θυμήθηκε ότι ο τελεστής `=` είναι ο τελεστής της ανάθεσης, ενώ `==` είναι ο τελεστής της ισότητας. Πιθανότατα, ήθελε να γράψει

```
x = 1
if x == 1:
    print('x is equal to 1')
```

3. Γράψτε ένα πρόγραμμα Python το οποίο τυπώνει τον μέγιστο των ακεραίων `x`, `y` και `z`. Πόσες συγκρίσεις κάνει το πρόγραμμά σας; Λάβετε υπ' όψιν σας όλες τις δυνατές περιπτώσεις.

Απάντηση. Ο κώδικας

```
M = x
if y > M:
    M = y
if z > M:
    M = z
print('Max is', M)
```

εκτελεί ακριβώς τρεις συγκρίσεις. Πέρα από αυτό, γενικεύεται εύκολα σε οποιοδήποτε πλήθος ακεραίων.

4. Ένα έτος είναι δίσεκτο αν διαρείται με το τέσσερα αλλά όχι με το 100, με εξαίρεση τα έτη που διαιρούνται με το 400. Γράψτε ένα πρόγραμμα Python που ελέγχει αν ένα έτος είναι δίσεκτο ή όχι.

Απάντηση. Σύμφωνα με τον ορισμό του δίσεκτου έτους, είναι αρκετό να γράψουμε

```

year = 2002
if year % 4 == 0 and year % 100 != 0 or year % 400 == 0:
    print('The year', year, 'is a leap year')
else:
    print('The year', year, 'is not a leap year')

```

Σημειώστε ότι δεν χρειάζονται παρενθέσεις γιατί ο τελεστής `and` έχει μεγαλύτερη προτεραιότητα από τον τελεστή `or`.

5. Γράψτε ένα πρόγραμμα Python το οποίο μετατρέπει μια βαθμολογία από την κλίμακα 0–100 σε βαθμολογία A έως F ως εξής: F για βαθμολογία στο διάστημα [0, 59], D για βαθμολογία στο διάστημα [60, 69], C για βαθμολογία στο διάστημα [70, 79], B για βαθμολογία στο διάστημα [80, 89], και A διαφορετικά.

Απάντηση. Θα μπορούσαμε να γράψουμε

```

score = 78
if score < 60:
    lettergrade = 'F'
elif score < 70:
    lettergrade = 'D'
elif score < 80:
    lettergrade = 'C'
elif score < 90:
    lettergrade = 'B'
else:
    lettergrade = 'A'

```

6. (Δύσκολη) Γράψτε ένα πρόγραμμα Python το οποίο τυπώνει τον μεγαλύτερο *περιττό* αριθμό μεταξύ των ακεραίων `x`, `y`, `z`, διαφορετικά το μήνυμα ότι όλοι οι αριθμοί είναι άρτιοι.

Απάντηση. Η λύση που ακολουθεί χρησιμοποιεί την ιδέα της άσκησης 3.

```

x = 12; y = 92; z = 13

if x % 2:
    largest = x
else:
    largest = None

if y % 2:
    if largest is None or y > largest:
        largest = y

if z % 2:
    if largest is None or z > largest:
        largest = z

if largest is not None:
    print('Largest =', largest)
else:
    print('All numbers are even')

```

7. Τι θα τυπώσουν οι παρακάτω εντολές;

```
print('Hello John')
print('Hello', 'John')
print('Hello' + 'John')
```

Απάντηση. Οι παραπάνω εντολές θα τυπώσουν

```
Hello John
Hello John
HelloJohn
```

8. Γράψτε ένα πρόγραμμα το οποίο θα ζητά από τον χρήστη να δώσει ένα όνομα και ένα επίθετο (χωριστά το καθένα). Μετά θα καταχωρεί το ονοματεπώνυμο σε μία μεταβλητή τύπου `str` στη μορφή: 'επίθετο, όνομα'. Τέλος, θα τυπώνει το αποτέλεσμα.

Απάντηση. Θα μπορούσαμε να γράψουμε

```
firstname = input('First name? ')
lastname = input('Last name? ')

fullname = lastname + ', ' + firstname
print(fullname)
```

9. Γράψτε κατάλληλες εντολές για την επεξεργασία της ακολουθίας χαρακτήρων `AppliedMathematics`. Συγκεκριμένα, γράψτε εντολές οι οποίες τυπώνουν:

- (α') Τους πρώτους 6 χαρακτήρες
- (β') Τους τελευταίους 5 χαρακτήρες
- (γ') Κάθε δεύτερο χαρακτήρα ξεκινώντας από τον τρίτο
- (δ') Όλους τους χαρακτήρες ξεκινώντας από το τέλος και προχωρώντας προς την αρχή
- (ε') Μια ακολουθία αποτελούμενη από τους χαρακτήρες στις άρτιες θέσεις ακολουθούμενη από τους χαρακτήρες στις περιττές θέσεις

Απάντηση. Γράφουμε,

```
s = 'AppliedMathematics'
print(s[:6]) # First 6 characters
print(s[-5:]) # Last 5 characters
print(s[2::2]) # Every second character starting from the third
print(s[::-1]) # Reversed
print(s[::2] + s[1::2]) # Even positions followed by odd positions
```

10. Γράψτε ένα πρόγραμμα το οποίο ζητάει από το χρήστη το όνομα και την ηλικία του, μετατρέπει την ηλικία του σε ημέρες και τυπώνει ένα κατάλληλο μήνυμα, π.χ. `John, you are 11344 days old`. Αγνοήστε τα δίσεκτα έτη.

Απάντηση. Γράφουμε,

```
name = input('Enter your name: ')
age = int(input('Enter your age: '))
print(name, 'you are', 365*age, 'days old!')
```

11. Γράψτε ένα πρόγραμμα το οποίο δέχεται μία συμβολοσειρά και τυπώνει την συμβολοσειρά αυτή χωρίς τον πρώτο και τον τελευταίο χαρακτήρα της. Για παράδειγμα, αν η αρχική συμβολοσειρά είναι `Nikos` τότε πρέπει να τυπωθεί `iko`.

Απάντηση. Γράφουμε,

```
s = input('Enter a string: ')
print(s[1:-1])
```

12. Γράψτε ένα πρόγραμμα το οποίο δέχεται δύο μονοψήφιους ακέραιους αριθμούς m , n και τυπώνει μία συμβολειρά η οποία αποτελείται από m φορές το ψηφίο m ακολουθούμενο από n φορές το ψηφίο n . Για παράδειγμα, αν $m = 2$ και $n = 4$ τότε θέλουμε να παραχθεί η '224444'.

Απάντηση. Γράφουμε,

```
m = input('Enter number m: ')
n = input('Enter number n: ')
print(int(m)*m + int(n)*n)
```