Τράπεζα Θεμάτων Python (Εισαγωγή – Επαναληπτικές Εντολές)

1) [Συμπλήρωση κώδικα] Μετατροπή τύπου για υπολογισμό συνόλου

Θέλουμε να ζητήσουμε από τον χρήστη πόσα τεμάχια θέλει και να υπολογίσουμε το συνολικό κόστος.

```
price = 3
qty = input('Πλήθος; ')
total = _____(qty) * price
print(total)
```

Θεωρία: Η input() επιστρέφει πάντα συμβολοσειρά (str). Για αριθμητικούς υπολογισμούς χρειάζεται μετατροπή τύπου: συνήθως int() για ακέραιους ή float() για δεκαδικούς. Αν δεν γίνει μετατροπή, προκύπτει σφάλμα τύπου.

- A. int
- B. float
- C. str
- D. και int kai float είναι σωστό

2) [Πολλαπλής επιλογής] Τι εκτυπώνει το range(start, stop, step);

Τι ακριβώς θα εμφανίσει το παρακάτω πρόγραμμα;

```
for i in range(2, 7, 2):
  print(i, end=' ')
```

Θεωρία: Η range(start, stop, step) ξεκινά από το start, σταματά πριν από το stop και προχωρά με βήμα step. Άρα με range(2, 7, 2) παίρνουμε τις τιμές 2, 4, 6.

- A. 2 4 6
- B. 2 4 6 8

- C. 2 3 4 5 6
- D. 3 5 7

3) [Πολλαπλής επιλογής] Εισαγωγική ερώτηση Python #3

Τι θα εμφανίσει ή ποια εντολή είναι σωστή στο παρακάτω απόσπασμα;

```
x = 3
if x > 2:
   print('Μεγαλύτερο')
else:
   print('Μικρότερο ή ίσο')
```

Θεωρία: Η δομή if χρησιμοποιείται για έλεγχο συνθηκών. Αν η συνθήκη είναι True, εκτελείται το αντίστοιχο μπλοκ εντολών.

- A. Θα εκτυπώσει Μεγαλύτερο
- B. Θα εκτυπώσει Μικρότερο ή ίσο
- C. Σφάλμα σύνταξης
- D. Δεν εκτυπώνει τίποτα

4) [Πολλαπλής επιλογής] Εκτύπωση αριθμών από 1 έως 5

Τι θα εμφανίσει το παρακάτω πρόγραμμα;

```
for i in range(1, 6):
   print(i, end=" ")
```

Θεωρία:
Η range(start, stop) παράγει ακολουθία από το start **μέχρι** (αλλά **όχι περιλαμβάνοντας**) το stop. Η παράμετρος end=" " κρατά την εκτύπωση στην ίδια γραμμή.

Επιλογές:
A. 1 2 3 4 5

B. 0 1 2 3 4 5
C. 1 2 3 4 5 6
D. 0 1 2 3 4

---

5) [Συμπλήρωση κώδικα] Άθροισμα αριθμών

Συμπλήρωσε το κενό ώστε ο κώδικας να υπολογίζει το άθροισμα των αριθμών 1 έως 5.

```
sum = 0

for i in _____(1, 6):

   sum += i

print(sum)
```

Θεωρία:
Η range(1, 6) παράγει τις τιμές 1, 2, 3, 4, 5. Η εντολή sum += i προσθέτει κάθε αριθμό στο άθροισμα.

Επιλογές:
A. range
B. list
C. for
D. int

6) [Πολλαπλής επιλογής] Εκτύπωση χαρακτήρων

Τι θα εμφανίσει το παρακάτω πρόγραμμα;

```
for ch in "Python":

   print(ch)
```

Θεωρία:
Η for μπορεί να επαναλάβει χαρακτήρες μιας συμβολοσειράς. Κάθε επανάληψη επιστρέφει έναν χαρακτήρα της ακολουθίας.

Επιλογές:
A. Όλα τα γράμματα της λέξης “Python” σε ξεχωριστές γραμμές
B. Την ίδια λέξη “Python” σε μία γραμμή
C. Μόνο το πρώτο γράμμα “P”
D. Τίποτα – θα προκαλέσει σφάλμα

7) [Πολλαπλής επιλογής] Τετράγωνα αριθμών

Τι θα εκτυπώσει το πρόγραμμα;

```
for i in range(3):
   print(i**2)
```

Θεωρία:
Ο τελεστής ** υψώνει έναν αριθμό σε δύναμη. Εδώ το i παίρνει τιμές 0, 1, 2.

Επιλογές:
A. 0 1 4
B. 1 4 9
C. 0 1 2
D. 1 2 3

8) [Συμπλήρωση κώδικα] Εμφάνιση μόνο ζυγών αριθμών

Συμπλήρωσε το κενό ώστε να εμφανίζονται μόνο οι ζυγοί αριθμοί από 0 έως 8.

```
for i in range(0, 10):
    if _________:
        print(i)
```

Θεωρία:
Η εντολή if μέσα σε for μπορεί να ελέγχει συνθήκες. Για ζυγούς αριθμούς, χρησιμοποιούμε τον τελεστή υπολοίπου %.

Επιλογές:
A. i % 2 == 0
B. i / 2 == 0
C. i * 2 == 0
D. i == 2

9) Τι θα εκτυπώσει το παρακάτω απόσπασμα κώδικα;

```
total = 0
for i in range(1, 5):
    if i % 2 == 0:
        total += 1                    // είναι ίδιο με total = total+1
print(total)
```

Hint: Ποιες τιμές παράγει η range(1, 5) και ποιες είναι ζυγές;

10) Ποια είναι η τελική τιμή της μεταβλητής result;

```
result = 1
for i in range(2, 7, 3):
    result *= i                 # είναι ίδιο με result = result*i
```

Hint: Δώστε προσοχή στο **βήμα** (step) της range().

11) Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
count = 0
for letter in "Python":
    if letter == 'P':
        count += 10
    elif letter == 'o':
        count += 5
print(count)
```

Hint: Η elif εκτελείται μόνο αν η if είναι ψευδής. Ο βρόχος επαναλαμβάνεται για κάθε χαρακτήρα.

12) Τι τιμή θα έχει η μεταβλητή count μετά την εκτέλεση των εμφωλευμένων βρόχων;

```
count = 0
for x in range(2):
  for y in range(2):
    if x != y:
      count += 1
```

Hint: Ποια ζεύγη τιμών (x, y) παράγονται και για ποια ισχύει η συνθήκη x != y;

13) Ποια είναι η τελική τιμή του x;

```
x = 0
for i in range(10):
  x += 1
  if i == 4:
    break
```

Hint: Η εντολή break τερματίζει τον βρόχο άμεσα. Μετρήστε τις αυξήσεις του x

14**) Τι θα εμφανίσει ο κώδικας;**

```
total = 0
for i in range(3):
  if i == 1:
    continue
  total += i
```

```
print(total)
```

Hint: Η continue παραλείπει την **τρέχουσα** επανάληψη.

15) Τι θα εκτυπωθεί;

```
s = 0
for num in range(3, 10):
    if num % 3 == 0:
        s += 1
print(s)
```

Hint: Βρείτε πόσα πολλαπλάσια του 3 υπάρχουν στο διάστημα από 3 έως 9 (συμπεριλαμβανομένου).

16) Τι θα εκτυπώσει το παρακάτω απόσπασμα;

```
for last_i in range(5):
    pass
if last_i < 4:
    print("Small")
else:
    print("Large")
```

Hint: Ποια είναι η **τελευταία** τιμή που παίρνει το last_i στον βρόχο;

17) Ποια θα είναι η τιμή της μεταβλητής word;

```
word = ""
my_string = "a_b_c"
for letter in my_string:
    if letter != '_':
        word += letter
```

Hint: Ποιοι χαρακτήρες του my_string **δεν** είναι κάτω παύλα;

18) Τι θα εμφανίσει ο κώδικας;

```
is_found = False
for i in range(5):
    if i == 3:
        is_found = True
if is_found:
    print("Y")
else:
    print("N")
```

Hint: Αλλάζει η τιμή της is_found σε True έστω και μία φορά στον βρόχο;

19) Τι θα εκτυπώσει ο κώδικας;

```
sum_val = 0
for i in range(1, 6):
    if i % 3 != 0:
        sum_val += i
print(sum_val)
```

Hint: Ποιους αριθμούς προσθέτουμε; Όχι τα πολλαπλάσια του 3.

20) Ποια είναι η τελική τιμή του count;

```
count = 0
for x in range(3):
    if x == 1:
        count += 2
    elif x == 2:
        count += 3
    else:
        count += 1
```

Hint: Η range(3) παράγει $0, 1, 2$. Υπολογίστε την αύξηση για κάθε τιμή x.

21) Τι θα εμφανίσει το πρόγραμμα;

```
message = ""
for char in "Hello":
    if char == 'l':
        message += '!'
    else:
        message += char
print(message)
```

Hint: Αντικαταστήστε τους χαρακτήρες 'l' με '!' στην αρχική λέξη.

22) Ποια είναι η τελική τιμή του result;

```
result = 10
for i in range(1, 4):
    if i == 2:
        break
    result += i
print(result)
```

Hint: Ο βρόχος σταματά όταν το i γίνει 2. Πόσες φορές προστίθεται το i στο result;

23) Τι θα εκτυπώσει ο κώδικας;

```
text = ""
for k in range(3):
    if k == 1:
        continue
    text += str(k)
print(text)
```

Hint: Η continue παραλείπει την επανάληψη για το $k=1$.

24) Τι θα εμφανίσει ο κώδικας;

```
found = False
for i in range(10, 20, 5):
    if i == 15:
        found = True
if found:
    print(i)
else:
    print(0)
```

Hint: Ποιες τιμές παράγει η range(10, 20, 5); Επίσης, ποια είναι η τελική τιμή της μεταβλητής i μετά τον βρόχο;

25) Ποια είναι η τελική τιμή του s;

```
s = 0
for x in range(5):
  if x % 2 == 0:
    for y in range(2):
      s += 1
```

Hint: Ο εξωτερικός βρόχος εκτελείται για τα $x=0, 2, 4$. Πόσες φορές εκτελείται συνολικά ο εσωτερικός βρόχος;

26) Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
score = 0
for val in [10, 5, 20, 8]:
  if val > 15:
    score += val
    break
  score += 1
print(score)
```

Hint: Η εντολή break εκτελείται μόλις βρεθεί η πρώτη τιμή >15.

27) Τι θα εκτυπωθεί;

```
result = ""
for i in range(1, 5):
    if i % 2 == 0:
        result += "Z"
    else:
        result += str(i)
print(result)
```

Hint: Οι μονοί αριθμοί γίνονται συμβολοσειρά, οι ζυγοί γίνονται "Z".

28) Τι θα εμφανίσει ο κώδικας;

```
flag = "Start"
for i in range(2):
    if i == 0:
        flag = "Middle"
    else:
        flag = "End"
    if flag == "End":
        print(flag)
```

Hint: Παρακολουθήστε την αλλαγή του flag. Πότε εκτελείται η print() και πότε αλλάζει το flag στην τιμή "End";