

Marmara Üniversitesi Teknik Bilimler Meslek Yüksekokulu
Algoritmalar ve Programlama
TEMEL KAVRAMLAR İÇİN ÇALIŞMA SORULARI

Soru 1:

```
A,B=23,10  
print(A-B)
```

a)11 b)-12 c)-13 d)13

Soru 2:

```
A,B=23,10  
if A%10>5:print(A-B)  
else:print(A+B)
```

a)2310 b)13 c)33 d)1023

Soru 3:

```
a=[10,20,30,40]  
print(a[1])
```

a)10 b)20 c)30 d)40

Soru 4:

```
a=[11,21,31,41]  
if a[2]>30:print(a[3])  
print(a[0])
```

a)41 ve 11 b)41 ve 21 c)41 d)11

Soru 5:

```
b=[[10,20],[30,40],[50,60]]  
print(b[2][1])  
a)10 b)30 c)40 d)60
```

Soru 6:

```
b=[[10,20],[30,40],[50,60]]  
c=b[1]  
print(c[1])
```

a)10 b)30 c)40 d)60

Soru 7:

```
x,y,z=1,2,3  
dizi=[x+y,x-y,x*y,x/y]  
print(max(dizi),min(dizi))
```

a)1 3 b)3 -1 c)-1 3 d)3 1

Soru 8:

```
dizi1= list(range(1,5))  
dizi2= list(range(5,1,-1))  
print(dizi1[0]+dizi2[3])
```

a)1 b)2 c)3 d)4

Soru 9:

```
for i in range(6,0,-2):  
    print(i,end=',')
```

a)6,5,4, b)2,4,6, c)6,4,2, d)2,4,

Soru 10:

```
for i in range(10):  
    if i==3:break  
print(i)
```

a)1 b)2 c)3 d)4

Soru 11:

```
for i in range(5):  
    if i==2:continue  
    print(i,end="")
```

a)0123 b)0134 c)134 d)124

Soru 12:

```
x,y=1,2  
while x<4:  
    x=x+y  
    print(x,end="")
```

a)23 b)34 c)35 d)36

Soru 13:

```
i=0  
while i<3:  
    i=i+1  
    print(i,end="")  
else:  
    print(i**2)
```

a)1239 b)3459 c)124 d)2416

Soru 14:

```
def islem(x,y):  
    if x>10:  
        return str(x)+"."+str(y)  
    else: return x+y
```

print(islem(11,23))

a)23 b)11.23 c)23.11 d)44

Soru 15:

```
def a(y):  
    x=y**2  
    return x%5  
print(a(13))
```

a)2 b)3 c)4 d)5

Soru 16:

```
def varmi(dizi,y):  
    if y in dizi:return "var"  
    else: return "yok"
```

```
sayilar=[20,30,40]  
print(varmi(sayilar,20))
```

a)2 b)3 c)var d)yok

Soru 17:

random.randint(x,y) verilen x ve y sayıları aralığında rastgele tam sayı üretir. abs(x) ise mutlak değer alır. Örnek=abs(-2) 2'dir. Buna göre:

```
import random  
x=random.randint(3, 6)-10  
print(abs(x))
```

a)1 b)2 c)4 d)8

Soru 18:

```
x=5 & 3  
print(x)
```

a)1 b)2 c)3 d)4

Soru 19:

```
x=5  
print(x>>1)
```

a)1 b)2 c)3 d)4

Soru 20:

```
a = "Gelmiyorlar"  
x=len(a)  
print(x,a[:3])
```

a)10 G b)11 Ge c)11 Gel d)10 Gel

Soru 21:

```
import math  
x=int(math.sqrt(16))  
print(x)
```

a)4.0 b)4 c)16 d)8

Değişkene değer atarken

a=10

b=20 yerine daha pratik olan

a,b=10,20 yöntemi de kullanılabilir.

% mod işlemi yapar. Örneğin x= 9%5 işlemi sonucu x 4 değerini alır. x=13%10 işlemi sonucu ise x 3 olur.

** kuvvet alma operatörüdür. 2**3 2 üzeri 3 demektir.

DİZİ KAVRAMLARI

notlar=[30,50,80] bir dizidir. notlar[0] dizinin ilk, notlar[2] son elemanıdır. Dizinin eleman sayısı **len** komutu ile bulunur x=len(notlar) sonucu burada x 3 değerini alır.

max ve **min** komutları dizinin en büyük ve en küçük elemanını verir.

notlar[1:] 1'den sonraki tüm elemanları verir. Yani [50,80]. notlar[:2] 2'ye kadar olan elemanları verir. Yani [30,50]

notlar=[[30,50],[80,90],[70,60]] iki boyutlu bir dizidir. notlar[0][0] ilk elemanın, yani [30,50], ilk elemanını verir. Yani 30.

range(10) 0-9 aralığıdır ama dizi değildir. **list** komutu ile diziye çevrilebilir. Bu şekilde **append** komutu kullanımı mümkün olur. notlar.append(3) gibi.

range(başlama,bitiş,adım sayısı) Örnek: range(1,9,2) 1'den 9'a kadar 2'şer. 1 dahil, 9 dahil değil. range(9,1,-2) ise 9'dan 1'e 2'şer azalarak gider.

DÖNGÜ KAVRAMLARI

continue döngüde bulunduğu satırdan aşağı devam etmez. Bir sonraki döngü adımına geçerek devam eder.

break döngüyü durdurur. Program döngü bloğu bitiminden sonraki satıra devam eder.

while bloğu içindeki kodlar koşul doğru olduğu müddetçe çalıştırılır. Koşul yanlış olduğunda **else:** içindeki kodlar bir kez çalıştırılarak döngü biter.

KÜTÜPHANE VE FONKSİYON

def fonksiyon tanımlamak için kullanılır. Her fonksiyondan bir değer döner. Def içerisinde **return** ile değer döndürülür.

Bir kütüphane import komutu ile çağrılır.

Örneğin **import math** matematik ile ilgili işlemler için çağrılır. Artık örneğin karekök almak için **math.sqrt** komutu kullanılabilir. Rastgele sayı üretmek için **random** kütüphanesi çağrılmalıdır.

random.randint(1,5) 1 ile 5 dahil, bu sayılar arasında rastgele bir sayı üretir. Bu durumda x=random.randint(1,5) komutu sonucu x 1,2,3,4,5 değerlerinden birini alabilir.

abs komutu mutlak değer alır. x=abs(-3) çalıştığında x 3 değerini alır.

BİT SEVİYESİ İŞLEMLER

& işareti bit seviyesinde mantıksal VE işlemi yapar. x=1 & 0 sonucu x 0 değeri alır. | ise mantıksal VEYA işlemi yapar.

x=4&5 işlemi şu şekilde yapılır:
100 4'ün ikilik karşılığı
101 5'in ikilik karşılığı
100 her bitin AND işlemi sonucu oluşan yeni sayı. Yani x burada 4 olur.

x=4|5 işlemi şu şekilde yapılır
100 4'ün ikilik karşılığı
101 5'in ikilik karşılığı
101 her bitin OR işlemi sonucu oluşan yeni sayı. Yani x burada 5 olur.

Python'da bit seviyesinde işlem yapan tüm operatörler aşağıdaki tabloda gösterilmiştir.

Operator	Örnek	Anlam
&	a & b	AND
	a b	OR
^	a ^ b	XOR
~	~a	NOT
<<	a << 3	left shift
>>	a >> 3	right shift

Kaydırma işlemi ise şu şekildedir.

x=4 bit seviyesinde 100'dür. x>>1 bir bit sağa kaydırma yapar. Sonuç bit seviyesinde 10 olur. Yani ondalık 2. x>>1 işlemi tekrar uygulanırsa 1 olur.

x=6 bit seviyesinde 110'dür. x>>1 bir bit sağa kaydırma yapar. Sonuç bit seviyesinde 11 olur. Yani ondalık 3.

x=6 bit seviyesinde 110'dür. x<<1 en sağa 0 ekleyerek bir bit sola kaydırma yapar. Sonuç bit seviyesinde 1100 olur. Yani ondalık 12.

Pratik olarak sağa kaydırma en sağdaki haneyi atma, sola kaydırmak en sağa 0 ekleme şeklinde düşünülebilir.

Kaydırma işlemi x>>3, x<<5 şeklinde farkı sayıda da yapılabilir.

GENEL BİLGİLER

print komutu parantez içinde verilen parametreleri ekrana yazar. Varsayılan durumda alt satıra atar. Bunu aslında \n karakteri kullanarak yapar. Alt satıra atmak yerine örneğin tab yapılmak istenirse

print(x,end='\t') kullanılır.

Örneğin

```
x="a"
print(x,end='')
print(x,end='')
```

ekrana aa yazar.

```
x="a"
print(x,end='_')
print(x,end='_')
```

Ekrana a_a_ yazar.