

```
>>> print(b)
```

```
6
```

b global o'zgaruvchiga asosiy dasturda murojaat qilinganda ham natija sifatida 5 ni emas 6 ni chiqaradi, sababi u funksiya ichida global sifatida e'lon qilingan.

O'zgaruvchi funksiya parametr sifatida

Agar o'zgaruvchi funksiya parametr sifatida ishlatilsa, uning qiymati lokal o'zgaruvchining yangi qiymati sifatida qabul qilinadi.

```
>>> def val4(d):
```

```
    print(d)
    d=100
```

```
    print(d)
>>> c=200
>>> val4(c)
```

```
200
100
```

d ning qiymati **val4** nomli funksiya qaysi qiymat bilan chaqirilsa, shu qiymatga teng, ya'ni *c* ga teng.

Funksiya ichida parametr sifatida kelgan *d* ning qiymatini ekranga chiqarish buyrug'i berildi.

Lokal o'zgaruvchi *d* ga yangi qiymat berildi.

d lokal o'zgaruvchining qiymati ekranga chiqarildi.

Global o'zgaruvchi *c* e'lon qilindi.

Asosiy dasturda **val4** nomli funksiya *c* parametr bilan chaqirilganda, birinchi parametr sifatida kelgan 200 ni, keyin esa lokal o'zgaruvchiga yangi berilgan qiymat 100 ni chiqaradi.



1. Funksiyaning parametrlari nima?
2. Funksiyaning tanasida qaysi kodlar yoziladi?
3. Lokal o'zgaruvchi deb nimaga aytiladi?
4. Global o'zgaruvchi deb nimaga aytiladi?
5. Lokal va global o'zgaruvchilarning farqi nimada?



1. *n* natural soni berilgan. $S=1*5+2*6+3*7+...+n*(n+4)$ ifodani hisoblash dasturini protsedura yordamida tuzing.
2. *a* va *b* natural sonlari berilgan. *a* va *b* sonlaridan kattasini topish funksiyasini tuzing. Funksiyadan foydalanib, *a*, *b* va *c* sonlari ichidan kattasini topish dasturini tuzing.

61-DARS. AMALIY MASHG'ULOT

Misol. Berilgan ikkita sonning eng katta umumiy bo'luvchisi (EKUB)ni topish dasturini tuzing. Funksiyadan foydalaning.

Berilgan ikkita sonning eng katta umumiy bo'luvchisi (EKUB)ni topishning eng oddiy usuli maktab darsliklarida ko'rsatib o'tilgan bo'lib, bunda ikkala son tub bo'luvchilarga ajratiladi va ulardan umumiyarlari ko'paytirib chiqiladi.

$$60=2*2*3*5$$

$$21=3*7$$

$$\text{EKUB}(60,21)=3$$

Bu usul matematikada qulay, lekin dasturlashda undan foydalanish noqulay va sekin kechadi. Shu sababli, **Yevklid algoritmi**dan foydalaniladi.

Yevklid algoritmidan 2 ta usuldan foydalaniladi.

1-usul. Ikki sonning kattasidan kichigini ayirib boriladi va ayirma kichik sonidan kichik bo'lib qolsa, ularning o'rnini almashtiriladi.

1) $60 - 21 = 39$; 2) $39 - 21 = 18$; 3) $21 - 18 = 3$.

$EKUB(60, 21) = 3$

2-usul. Qadamlar sonini kamaytirish uchun ayirish o'rniga qoldikli bo'lishdan foydalanish mumkin.

1. Ikki sonning kattasini kichigiga bo'lib, qoldiq topiladi.

2. Ularning o'rnini almashtiriladi.

3. 1- va 2-qadamlar sonlardan biri nol bo'lguncha davom ettiriladi.

4. Oxirida qolgan son shu ikki son uchun EKUB bo'ladi.

$60 / 21 = 2 * 21 + 18$ qoldiq

$21 / 18 = 1 * 18 + 3$ qoldiq

$18 / 3 = 6 * 3 + 0$ qoldiq

$EKUB(60, 21) = 3$

<pre>def ekub(a, b): while a != 0 and b != 0: if a > b: a %= b else: b %= a ekub_q = a + b return ekub_q a = int(input()) b = int(input()) print(ekub(a, b)) 60 21 3</pre>	a, b parametrlarni qabul qiluvchi ekub nomli funktsiya e'lon qilindi.
	Sikl tanasi buyruqlari a va b 0 ga teng bo'lguncha takrorlanadi: 1-sikl. $a = 60; b = 21$ 2-sikl. $a = 18; b = 21$ 3-sikl. $a = 18; b = 3$ 4-sikl. $a = 0; b = 3$ sikldan chiqadi.
	Agar a soni b dan katta bo'lsa, a/b qoldig'ini a ga o'zlashtiradi: 1-sikl. $60 > 21 \rightarrow a = 60 \% 21 = 18$ 3-sikl. $18 > 3 \rightarrow a = 18 \% 3 = 0$
	Aks holda, b/a qoldig'ini b ga o'zlashtiradi: 2-sikl. $18 < 21 \rightarrow b = 21 \% 18 = 3$
	$ekub_q = 0 + 3 = 3$ EKUB ni hisoblaydi.
	ekub_q qiymatni qaytaradi.
	a, b ga qiymat beriladi.
	2 ta son EKUBining funktsiyasi chaqirilib, ekranga natija chiqariladi.

1. Berilgan to'rtta sonning eng kichigini topuvchi dastur tuzing. Buning uchun $min4(a, b, c, d)$ funktsiyasini yarating.

2. Berilgan a (haqiqiy) sonining k (butun) darajasini topuvchi dastur tuzing. Buning uchun $daraja(a, k)$ funktsiyasini yarating.

3. Inglizcha harf va raqamlardan iborat satr berilgan. Ushbu satr qancha raqamdan iboratligini hisoblovchi dastur tuzing.

