INFORMATICĂ · PYTHON · CLASA A VII-A · STRUCTURA ALTERNATIVĂ · LECȚIA 9

elif și condiții compuse: and, or, not

Durată: 50 min Nivel: gimnaziu · Python Înainte: lecțiile py7g-08

CE VEI ȘTI DUPĂ LECȚIE

- Alegi între mai mult de două variante cu if – elif – else.
- Combini condiții cu and, or și not.
- Rezolvi probleme precum maximul a trei numere sau încadrarea într-un interval.

1. Mai multe drumuri: elif

Uneori nu ajung două variante. Nota de la școală, de exemplu, se transformă în patru calificative. Pentru asta folosim `elif` (de la „else if", adică **altfel, dacă**).

```
nota = int(input())
if nota >= 9:
    print("Foarte bine")
elif nota >= 7:
    print("Bine")
elif nota >= 5:
    print("Suficient")
else:
    print("Insuficient")
```

Pentru intrarea `8`:

```
Bine
```

Python verifică condițiile **de sus în jos**. La prima condiție adevărată execută blocul ei și **sare peste restul**. `else` de la sfârșit prinde tot ce n-a nimerit în niciuna dintre variante. Poți pune câte `elif`-uri ai nevoie.

ATENȚIE — GREȘELI FRECVENTE

Ordinea condițiilor contează! Ce se întâmplă dacă începi cu cea mai mică notă?

```
if nota >= 5:
    print("Suficient")
elif nota >= 9:
    print("Foarte bine")
```

Pentru nota `9`, programul afișează `Suficient`, pentru că prima condiție este deja adevărată, iar `elif`-ul nu mai este citit niciodată. Pune întâi condițiile **cele mai stricte**.

2. Condiții compuse: and, or, not

Cu trei cuvinte poți **lipi** mai multe condiții într-una singură:

Cuvânt	Înseamnă	Este adevărat când...
`and`	**și**	**ambele** condiții sunt adevărate
`or`	**sau**	**cel puțin una** este adevărată
`not`	**nu**	condiția este **falsă** (o răstoarnă)

Exemple cu `x = 15`:

```
x = 15
print(x > 10 and x < 20)
print(x < 5 or x > 12)
print(not x > 10)
```

```
True
True
False
```

- `x > 10 and x < 20`: 15 este mai mare ca 10 **și** mai mic ca 20 → `True`;
- `x < 5 or x > 12`: prima parte e falsă, dar a doua e adevărată, iar la `or` ajunge una → `True`;
- `not x > 10`: `x > 10` este adevărat, iar `not` îl răstoarnă → `False`.

3. Exemple cu decizii

Un număr de două cifre este între 10 și 99:

```
n = int(input())
if n >= 10 and n <= 99:
    print("are două cifre")
else:
    print("nu are două cifre")
```

Pentru intrarea `47`:

```
are două cifre
```

Bilet la cinema. Intrarea este gratuită pentru copii sub 6 ani **sau** pentru persoane peste 65 de ani:

```
v = int(input())
if v < 6 or v > 65:
    print("gratuit")
else:
    print("5 lei")
```

Pentru intrarea `70`:

```
gratuit
```

> **DE ȘTIUT**
>
> Python permite și scrierea din matematică: `10 <= n <= 99`. Merge la fel ca `n >= 10 and n <= 99`.

4. Maximul a trei numere

Un număr este maxim dacă este mai mare sau egal cu **amândouă** celelalte, adică avem nevoie de `and`:

```
a = int(input())
b = int(input())
c = int(input())
if a >= b and a >= c:
    print(a)
elif b >= a and b >= c:
    print(b)
else:
    print(c)
```

Pentru intrările `4`, `9`, `6`:

```
9
```

Dacă `a` nu este maximul și nici `b` nu este, atunci maximul este sigur `c`: de aceea ultima variantă este un simplu `else`.

5. Verifică-te

1. Ce afișează programul pentru `x = 3` și pentru `x = 12`?

```
if x < 5:
    print("A")
elif x < 10:
    print("B")
else:
    print("C")
```

2. Scrie o condiție care este adevărată doar când `n` este între 1 și 9, inclusiv.
3. Când este adevărat `a > 0 or b > 0`? Dar `a > 0 and b > 0`?
4. Cum scrii „`n` NU este divizibil cu 3” în două feluri?

Răspunsuri

1. Pentru `x = 3`: `A`. Pentru `x = 12`: `C`.
2. `n >= 1 and n <= 9` (sau `1 <= n <= 9`).
3. Prima: când **cel puțin unul** dintre numere este pozitiv; a doua: când **ambele** sunt pozitive.
4. `n % 3 != 0` sau `not n % 3 == 0`.

Probleme de lucru în clasă (pbinfo)

Le rezolvăm împreună în ședință, în ordinea de mai jos.

- [ ] #9 **maxmin** — ușoară
  https://www.pbinfo.ro/probleme/9/maxmin

Temă pentru acasă (pbinfo)

Trimite soluțiile pe pbinfo și notează punctajul obținut. Bifează ce ai terminat.

- [ ] #106 **minim3** — ușoară
  https://www.pbinfo.ro/probleme/106/minim3