INFORMATICĂ · PYTHON · CLASA A VIII-A · TABLOURI UNIDIMENSIONALE · LECȚIA 5

# Minim și maxim, cu poziții

Durată: 50 min  Nivel: gimnaziu · Python  Înainte: lecțiile py8g-04

**CE VEI ȘTI DUPĂ LECȚIE**

- Găsești cel mai mare și cel mai mic element dintr-un șir.
- Ții minte și poziția (indicele) lui.
- Numeri de câte ori apare maximul.

## 1. Cine a sărit cel mai mult?

Cinci copii sar în lungime: 210, 185, 230, 199, 230 cm. Cine a sărit cel mai departe? Te uiți pe rând și ții minte **cel mai bun rezultat de până acum**. Când găsești unul mai bun, îl înlocuiești. Așa funcționează algoritmul pentru maxim.

## 2. Maximul unui șir

```
v = [210, 185, 230, 199, 230]
mx = v[0]
for x in v:
    if x > mx:
        mx = x
print(mx)
```

```
230
```

Pornim cu `mx = v[0]` : primul element este deocamdată „campionul”. Apoi îl comparăm cu toți ceilalți.

| `x` | `x > mx` ? | `mx` după pas |
|---|---|---|
| 210 | nu | 210 |
| 185 | nu | 210 |
| 230 | da | 230 |
| 199 | nu | 230 |
| 230 | nu | 230 |

**ATENȚIE — GREȘELI FRECVENTE**

Nu porni cu `mx = 0` ! Dacă toate numerele sunt negative (de exemplu `-5 -3 -8` ), ai afișa `0` , deși `0` nu face parte din șir. Pornește mereu cu primul element.

## 3. Minimul

La minim schimbăm doar semnul comparației: `<` în loc de `>`.

```
v = [210, 185, 230, 199, 230]
mn = v[0]
for x in v:
    if x < mn:
        mn = x
print(mn)
```

```
185
```

## 4. Poziția maximului

Cine a sărit cel mai departe? Ne trebuie **indicele**. Parcurgem după `i` și ținem minte poziția campionului.

```
v = [210, 185, 230, 199, 230]
poz = 0
for i in range(len(v)):
    if v[i] > v[poz]:
        poz = i
print("maxim", v[poz], "pe poziția", poz)
```

```
maxim 230 pe poziția 2
```

Valoarea 230 apare de două ori (pozițiile 2 și 4), dar noi am păstrat **prima** poziție. Cu `>` ne oprim la primul maxim. Dacă am scrie `>=`, am păstra **ultima** poziție (4).

## 5. De câte ori apare maximul?

Mai întâi găsim maximul, apoi parcurgem încă o dată și numărăm.

```
v = [210, 185, 230, 199, 230]
mx = v[0]
for x in v:
    if x > mx:
        mx = x
c = 0
for x in v:
    if x == mx:
        c += 1
print(mx, c)
```

```
230 2
```

SFAT

**Scurtătură (opțional).** Python are `max(v)` și `min(v)`. Ele fac exact ce am scris noi mai sus. Algoritmul tot trebuie să-l știi, pentru că la probleme ți se cere uneori și poziția.

## 6. Verifică-te

1. Ce valoare dă `mx` la final, pentru `[4, 9, 2, 9]`?
2. De ce nu pornim maximul de la `0`?
3. Cum schimbi programul pentru minim?
4. Pentru `v = [4, 9, 2, 9]`, ce poziție găsește bucla cu `>`? Dar cu `>=`?
5. Scrie condiția care numără câte elemente sunt egale cu minimul `mn`.

### Răspunsuri

1. `9`.
2. Pentru că șirul poate conține doar numere negative, iar `0` nu e în șir.
3. Pornești cu `mn = v[0]` și compari cu `<`.
4. Cu `>`: poziția 1 (prima apariție). Cu `>=`: poziția 3 (ultima).
5. `if x == mn: c += 1`.

## Probleme de lucru în clasă (pbinfo)

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

- [ ] #354 **n-maxim** — ușoară
  https://www.pbinfo.ro/probleme/354/n-maxim

## Temă pentru acasă (pbinfo)

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

- [ ] #346 **maxandap** — ușoară
  https://www.pbinfo.ro/probleme/346/maxandap