INFORMATICĂ · CLASA A VI-A · ANIMAȚII GRAFICE ȘI PROIECT · LECȚIA 6

Animații și proiect final: un joc cu repetiții

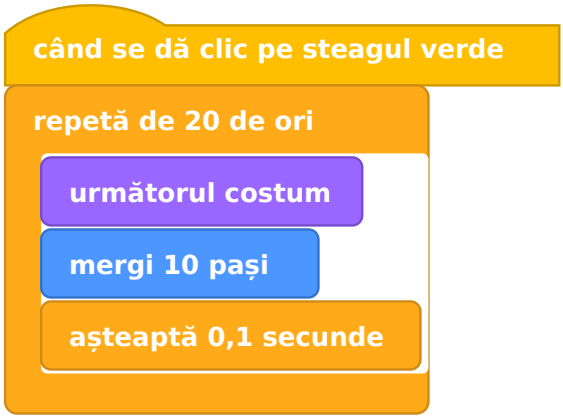
Durată: 60 min Nivel: gimnaziu Înainte: lecțiile g6-05

CE VEI ȘTI DUPĂ LECȚIE

- Creezi o animație schimbând costumele unui personaj într-o repetiție.
- Controlezi animația de la tastatură.
- Folosești „la nesfârșit" pentru jocurile care rulează continuu.
- Construiești un joc complet cu repetiții, decizii și variabile.

1. Animație = imagini schimbate repede

Un personaj „merge" dacă îi schimbi costumul (poza) de mai multe ori pe secundă și îl muți puțin de fiecare dată:



Animație

Poți desena sau modifica propriile costume în tab-ul **Costume** (editorul grafic din Scratch).

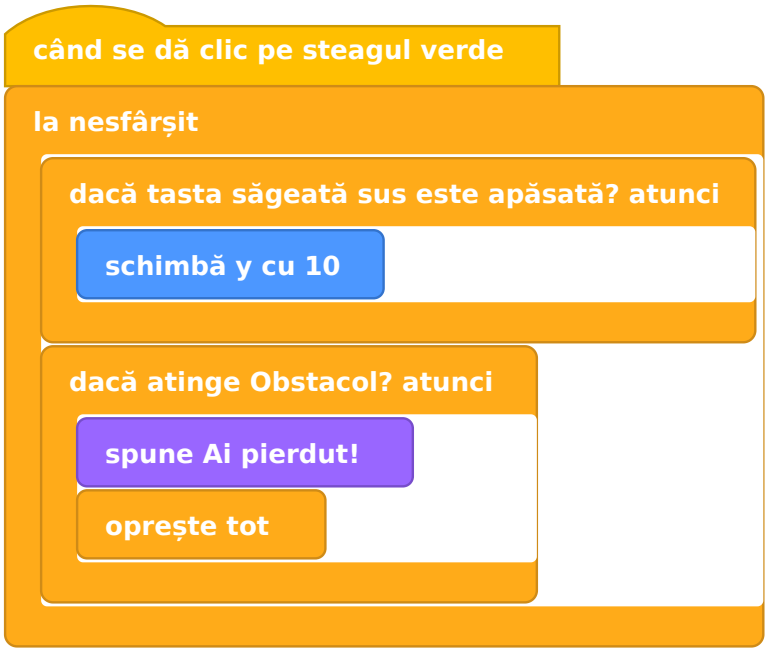
Repetare	Costum	Poziția x după pas
1	2	+10
2	1	+20
3	2	+30
…	…	…
20	1	+200

DE CE AȘA?

De ce o repetare cu număr fix aici? Știm exact câți pași vrem (20): structura `repetă de n ori`. Dacă nu știm (personajul merge până atinge marginea), folosim `repetă până când`.

2. Repetiția „la nesfârșit”

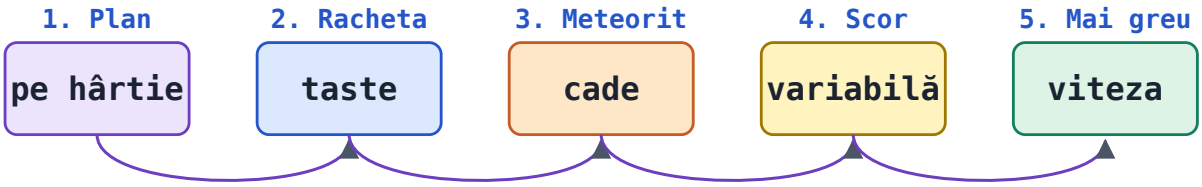
Jocurile rulează până le oprești. Blocul `la nesfârșit` (Control) repetă mereu; combinat cu `dacă`, verifică permanent ce se întâmplă:



La nesfârșit

> **DE CE AȘA?**
>
> **De ce `la nesfârșit` și `dacă`, nu doar `când tasta … este apăsată`?** Blocul `când tasta` pornește scriptul o dată la apăsare (cu o mică întârziere la repetare). Cu `la nesfârșit` + `dacă tasta … este apăsată` programul verifică la **fiecare cadru** (~30 de ori pe secundă) și mișcarea e lină.

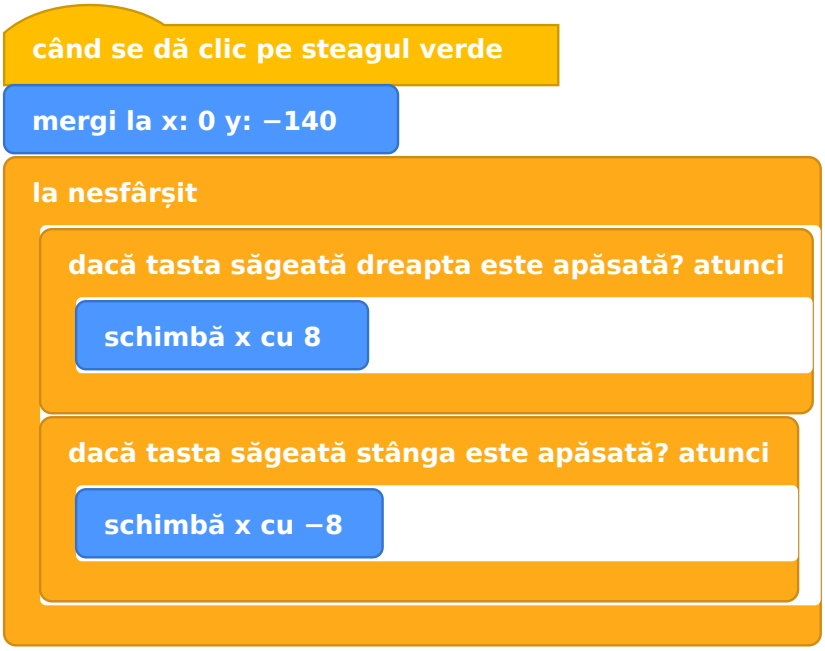
3. Proiect: „Fuga de meteoriți”



Etapele

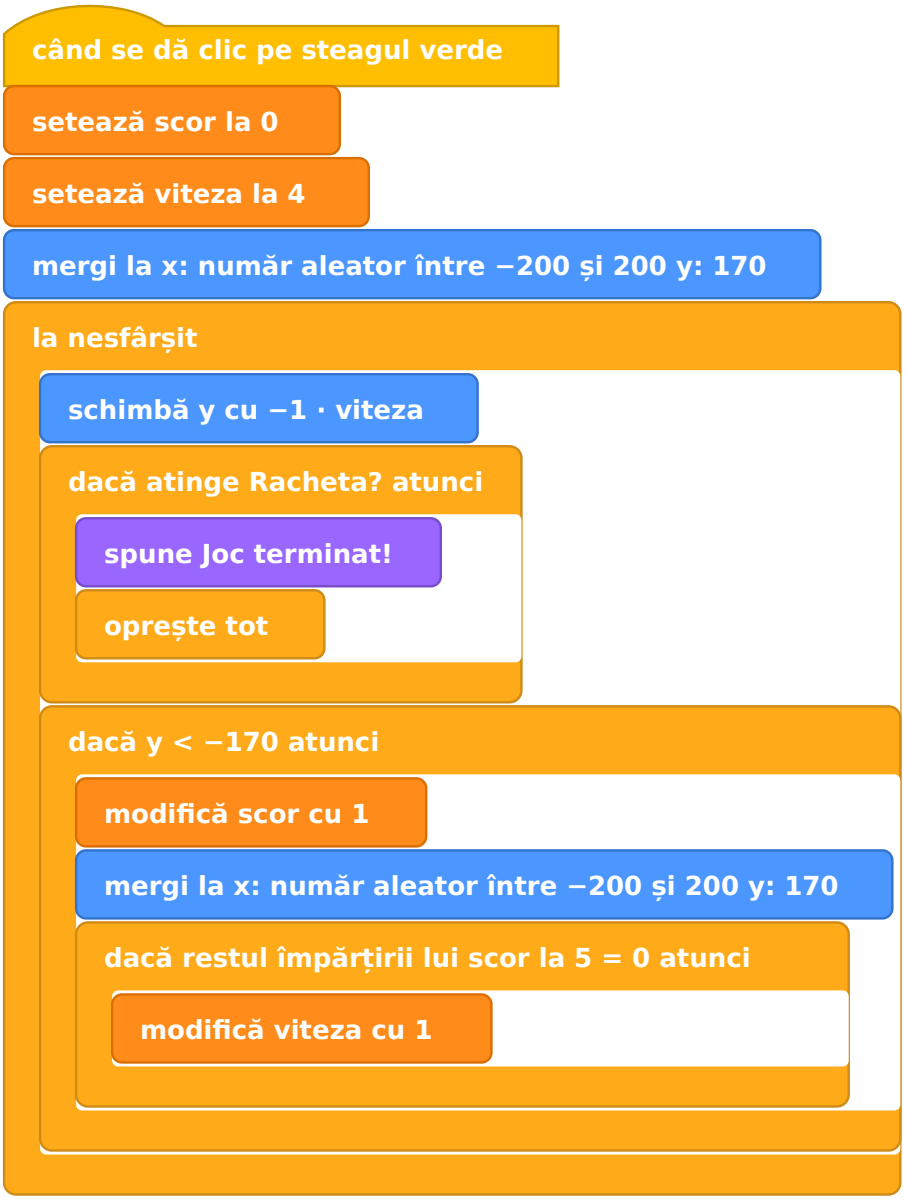
- **Racheta** se mișcă stânga–dreapta cu săgețile (la nesfârșit + dacă).
- **Meteoritul** cade de sus, repetat; când ajunge jos, reapare sus într-un loc aleator și scorul crește cu 1.
- Dacă meteoritul atinge racheta → „Joc terminat”.
- Bonus: la fiecare 5 puncte, meteoritul cade mai repede (variabila `viteza`).

Pasul 1: racheta.



Racheta

Pasul 2: meteoritul, scorul și viteza.



Meteoritul

Urmărim primele valori (scor și viteză):

Eveniment	scor	viteza
început	0	4
meteoritul cade 1 dată jos	1	4
a 2-a, a 3-a, a 4-a	4	4
a 5-a (5 mod 5 = 0)	5	5
a 10-a	10	6

DE CE AȘA?

De ce `scor mod 5 = 0`? Restul împărțirii la 5 este 0 pentru 5, 10, 15, ...: fiecare multiplu de 5. Condiția se aplică odată cu creșterea scorului, deci viteza crește exact la fiecare 5 puncte.

> **SFAT**
>
> Fă proiectul pe etape (racheta → meteorit → scor → viteză) și testează după fiecare. Dacă ceva nu merge, știi că problema e în ultima bucată adăugată.

4. Fișa de verificare

Am folosit...	Da / Nu
o repetiție cu număr fix de pași	
o repetiție la nesfârșit sau până când	
cel puțin o decizie (dacă)	
cel puțin două variabile (scor, viteză)	
o animație cu costume	
un sunet sau un fundal ales de mine	

5. Greșeli frecvente în proiecte

Problema	Cauza probabilă
racheta nu răspunde la taste	scriptul nu este în `la nesfârșit`
scorul crește foarte repede	condiția `y < -170` rămâne adevărată mai multe cadre (mută meteoritul sus imediat)
meteoritul trece prin rachetă	`atinge Racheta?` în afara buclei sau nu e verificat deloc
jocul nu se oprește la final	lipsește `oprește tot`
viteza nu mai crește	condiția `scor mod 5 = 0` este în afara blocului în care scorul crește

6. Exersează pe hârtie

1. Meteoritul pornește de la y = 170 cu viteza 4 (scade 4 pe cadru). La ce y se află după 10 cadre? După 40?

2. Câte cadre îi trebuie ca să ajungă sub y = −170, cu viteza 4? (aproximativ) Dar cu viteza 8?

3. Pentru ce valori ale scorului crește viteza, dacă regula este `scor mod 5 = 0`, de la 1 la 20?

4. Ce se întâmplă dacă ștergi `oprește tot` din scriptul de final?

5. Scrie în pseudocod regula „jocul se termină dacă meteoritul atinge racheta sau scorul ajunge la 30".

6. Desenează schema logică pentru: „la nesfârșit: dacă meteoritul atinge racheta → Joc terminat".

7. O animație are 3 costume și schimbă costumul de 12 ori. Ce costum se afișează la a 12-a schimbare (începe cu costumul 1, ciclic)?

8. De ce o repetare `la nesfârșit` fără nicio condiție de oprire nu este o greșeală într-un joc, dar este o greșeală într-un algoritm?

7. Mini-grile

1. `la nesfârșit` repetă: a) o dată b) de 10 ori c) până când oprești programul d) niciodată

2. Animația unui personaj se face prin: a) schimbarea fundalului b) schimbarea rapidă a costumelor c) schimbarea numelui d) ștergerea scriptului

3. `scor mod 5 = 0` este adevărată pentru scorul: a) 3 b) 5 c) 7 d) 11

4. Ce face blocul `oprește tot`? a) oprește doar scriptul b) oprește toate scripturile c) închide Scratch d) șterge variabilele

8. Mai departe (pentru curioși)

Clone. Blocul `creează un clon al meu` face mai mulți meteoriți în același timp, fără copiere manuală: fiecare clon are propriul script `când pornesc ca un clon`.

Mai multe vieți. Adaugă o variabilă `vieți` (3 la început); la fiecare lovitură scade cu 1; jocul se termină când `vieți = 0`.

Cel mai bun scor. Un al doilea variabil `record`: dacă `scor > record`, `setează record la scor`. Exact modelul „maximul" din lecția precedentă!

Spre liceu. `la nesfârșit` + `dacă` este „bucla principală a jocului" (game loop) din orice joc real: citește tastele, mută obiectele, verifică coliziunile, desenează, repetă. În Python se numește `while True`.

9. Ce am învățat

- Animația = schimbare de costume + pauze scurte, într-o repetiție.
- `la nesfârșit` + `dacă` = inima oricărui joc.
- Variabilele (scor, viteză) fac jocul interesant; `mod` ne ajută să vedem multipli.
- Un proiect bun se construiește pe bucăți, testate pe rând.

Citește și exersează în plus

- Scratch — editorul online — aici rulezi exemplele din lecție https://scratch.mit.edu/projects/editor/
- Scratch Wiki — Variables — cum folosești variabilele în programe (engleză) https://en.scratch-wiki.info/wiki/Variables
- Code.org — Hour of Code — puzzle-uri de programare cu blocuri https://code.org/hourofcode/overview

Răspunsuri

Exersează pe hârtie

1. După 10 cadre: 170 − 4·10 = **130**. După 40: 170 − 160 = **10**.

2. Distanța 340: cu viteza 4, 340 / 4 = 85 de cadre (și încă unul ca să fie *sub* −170, deci 86). Cu viteza 8: 340 / 8 = 42,5, deci **43** de cadre.

3. Pentru scorurile **5, 10, 15, 20**.

4. Jocul continuă după „Joc terminat": meteoritul și racheta se mișcă în continuare, scorul poate crește.

5. `dacă meteoritul atinge racheta sau scor = 30 atunci spune "Joc terminat" și oprește jocul sfârșit dacă`.

6. START → „la nesfârșit" (buclă) → romb „atinge racheta?" → DA: „Joc terminat" (STOP); NU: înapoi la începutul buclei.

7. Schimbările ciclice: 1→2, 2→3, 3→1 etc. După 12 schimbări (12 mod 3 = 0) se revine la costumul **1**.

8. Un joc trebuie să ruleze până îl oprește jucătorul; un algoritm trebuie să fie **finit** (se termină), altfel nu dă niciodată un rezultat.

Mini-grile

1. c. **2.** b. **3.** b. **4.** b.