

Descoperiți sau ilustrați concepte de codare cu un joc de puzzle distopic

Etape obligatorii anterioare / Cunoștințele anterioare ale elevilor	bază ale codării, în caz contrar este necesară o bună supraveghere a profesorului
Obiective de învățare	Descoperirea programării încă din primele niveluri. Suplimentarea unei ore de curs despre concepte de codare.
Subiecte	Alfabetizare digitală, Programare, Matematică
Vârsta recomandată	15-18
Materiale necesare	Joc: 7 miliarde de oameni (Windows, MacOs, Linux, AppStore, GooglePlay)
Durata secvenței	60 de minute
Activitate individuală sau de grup	grup
Abilități dezvoltate (după obiectivele de învățare)	Rezolvarea problemelor, planificare
Gama de prețuri al jocului	<15 €, <6 € (aplicații mobile)
Jocuri similare pe care le puteți utiliza cu secvența	Human Resource Machine (https://tomorrowcorporation.com/humanresource-machine)



<p>Sfaturi pentru o durată mai scurtă</p>	<p>Pentru a scurta această secvență, puteți străbate primele niveluri și să înțelegeți bine conceptele de joc pe care doriți să le arătați elevilor dvs. Apoi, în loc să treceți prin acele nivele cu ei, puteți trece direct la un nivel superior și le puteți explica toate conceptele într-o manieră de încercare și eroare („cum am putea face asta?”), apoi, folosind funcția de depanare, puteți arăta greșelile lor și încercați să le corectați până la succes.</p>
<p>Sfaturi pentru a face secvențele mai accesibile sau mai incluzive</p>	<p>Jocul nu este disponibil în limba greacă. Jocul nu conține prea mult vocabular în limba engleză. Ar trebui să oferiți singur explicațiile în loc să citiți textele în limba engleză elevilor dvs. De asemenea, ar trebui să explicați semnificația comenzilor („preluare”...).</p>



Pas cu pas: cum să implementezi secvența

În această secvență vei juca un joc puzzle de programare împreună cu clasa ta. Jocul folosește umor negru și un cadru drăguț, dar distopic, făcându-l o satiră ușoară a muncii moderne de birou. Progresând prin niveluri de dificultate crescândă, vei descoperi noi concepte de programare. Primele niveluri sunt destul de simple, dar va trebui să le supraveghezi și să oferi mai multe sfaturi pentru cele ulterioare.

Ar fi mai bine să pregătești sesiunea jucând singur prin primele 10 niveluri. Acest lucru vă va permite să treceți la niveluri mai provocatoare și să evitați explicațiile jocului dacă abilitățile elevilor dvs. sunt suficiente.

- **Pasul 1 – Prezentarea jocului și a secvenței (5 minute)**

Spune-le elevilor tăi că vei juca un joc puzzle pentru a aborda principiile de programare în mod jucăuș. Dacă ați prezentat deja concepte de programare în clasă, utilizați-l ca o modalitate de a le ilustra într-un mod amuzant.

Lansați și începeți jocul. Vă va cere să selectați un ID de angajat (adică un slot pentru salvarea jocului) și un avatar. Puteți trece rapid prin acest proces.

Este redată o secvență cinematografică care prezintă universul satiric al jocului, urmată de ecranul de selecție a nivelului.

Alegeți următorul nivel („Bine ați venit noilor angajați”).



- **Pasul 2 – Demonstrarea mecanicii jocului (10 minute)**

Explicați interfața și mecanica jocului parcurgând primele 3 niveluri.

Pentru acest pas, efectuați singur acțiunile și comentați-le. Asigurați-vă că elevii dvs. înțeleg mecanica.

Defalcarea interfeței:

Mica notă galbenă explică obiectivul dvs.:



Captură de ecran în joc: mecanica de bază pasul 1

Cutia cu comenzi de programare se află chiar lângă ea.

Când plasați comenzi în fereastra de sub ele, vi se poate cere să alegeți câțiva parametri:



Captură de ecran în joc: mecanică de bază pasul 2

Fiecare pas-comandă corespunde unei plăci de podea din camera muncitorilor.



Executați programul făcând clic pe butonul de redare din colțul din stânga jos:



Captură de ecran în joc: executarea programului



Nivelul 4 („Livrare la distanță lungă”) prezintă conceptul de bucle:



Captură de ecran în joc: introducerea buclelor

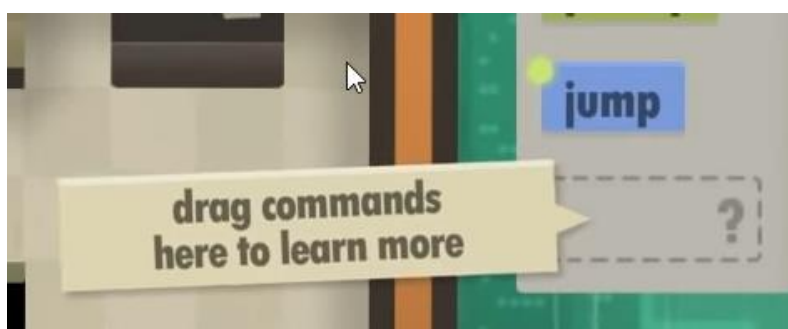
Joacă cu clasa ta într-un mod interactiv de la nivelul 4 încolo (pasul următor).



- **Pasul 3 – Treceți prin niveluri cu clasa dvs. (35 de minute)**

Pentru fiecare nivel:

- Spuneți clar obiectivul nivelului
- Explicați noile comenzi pe care jocul le-a pus la dispoziție pe măsură ce progresați:
 - Explicați cum funcționează folosind slotul de ajutor din partea de jos a lista de comenzi



- Combinați-le cu alte comenzi pentru a demonstra cum funcționează prin rularea unui cod de testare neterminat.
- Acordați elevilor câteva minute pentru a se gândi la o soluție, apoi întrebați cine dorește să o încerce și să implementeze soluția unui student.
 - Dacă funcționează, grozav! Asigurați-vă că toată lumea înțelege înainte de a merge mai departe.
 - Dacă nu, intrați în modul de depanare și încercați să înțelegeți ce a mers prost.



Procesul de depanare este explicat la nivelul 6 („micul exterminator 1”). Puteți rula codul linie cu linie folosind butonul săgeată:



Dacă elevii dvs. se luptă, descompuneți pașii pentru ei și întrebați-i ce comandă trebuie utilizată pentru fiecare pas.

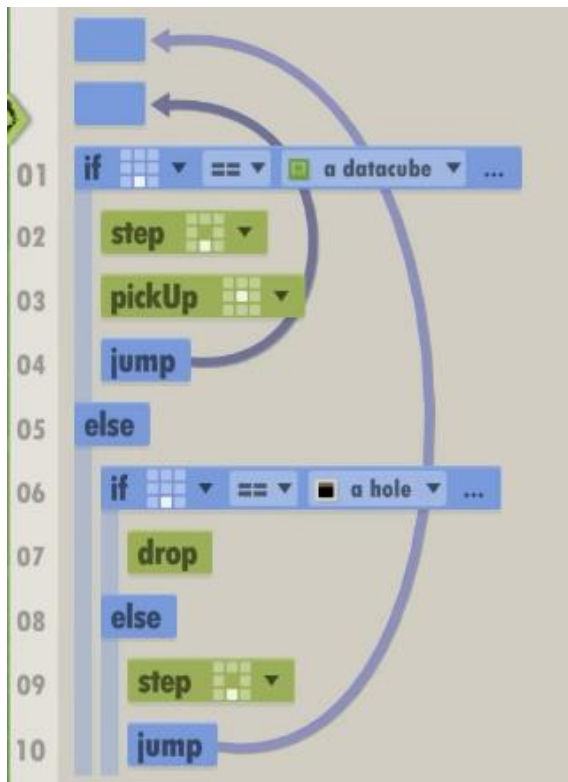
Dacă sunt complet blocați, dați răspunsul. A aștepta prea mult timp ar distruge dinamismul sesiunii. Plăcerea de a primi soluția unei probleme poate fi la fel de plăcută ca și să o rezolvi singur (gândește-te la romanele de mister). Cu toate acestea, asigurați-vă că elevii înțeleg complet soluția.



- **Pasul 4 – Debrief și uvertură (10 minute)**

Explicați conceptul de cutie neagră în programare: funcționarea interioară a unei funcții nu trebuie să fie cunoscută pentru a o utiliza.

Comparați un algoritm în Python cu un extras de algoritm din joc și comparați-le.



```
x = 0
y = 0
tile = workspace.get(x,y)
character = Character(tile, workspace)

while(True):
    if(isinstance(workspace.get(character.getCurrentTile().getBottom()), "Datacube")):
        character.step(character.getCurrentTile().getBottom())
        character.pickup()

    else:
        if(isinstance(workspace.get(character.getCurrentTile().getBottom()), "Hole")):
            character.drop()
        else:
            character.step(character.getCurrentTile().getBottom())
```

Acestea sunt reprezentări (limbi) diferite ale aceluiași algoritm. Nu cunoaștem codul din interiorul funcțiilor, dar tot ce trebuie să știm este ce acțiune efectuează. În acest fel le putem aranja pentru a produce un algoritm care se potrivește nevoilor noastre.



Resurse

Obținerea jocului

https://store.steampowered.com/app/792100/7_Billion_Humans/

<https://www.epicgames.com/store/fr/p/7-billion-humans>

<https://apps.apple.com/fr/app/7-billion-humans/id1393923918>

<https://play.google.com/store/apps/details?id=com.tomorrowcorporation.sevenbillionhumans>

Informații

Guevara, WG (2018, 29 noiembrie). Ce este „codul cutie neagră” și de ce este important. thatsoftwaredude.com Preluat de la <https://www.thatsoftwaredude.com/content/8881/what-is-black-box-code-and-why-its-important>.

Bristol, JB (2019). 7 miliarde de oameni: puzzle-ul amuzant provoacă copiii, învață principiile de programare. commonsense.org Preluat de la <https://www.commonsense.org/education/app/7-billion-humans>.

Toate capturile de ecran folosite în această lecție au fost preluate de la 7 Billion Human, Tomorrow Corporation (2018)

