

Tema 2

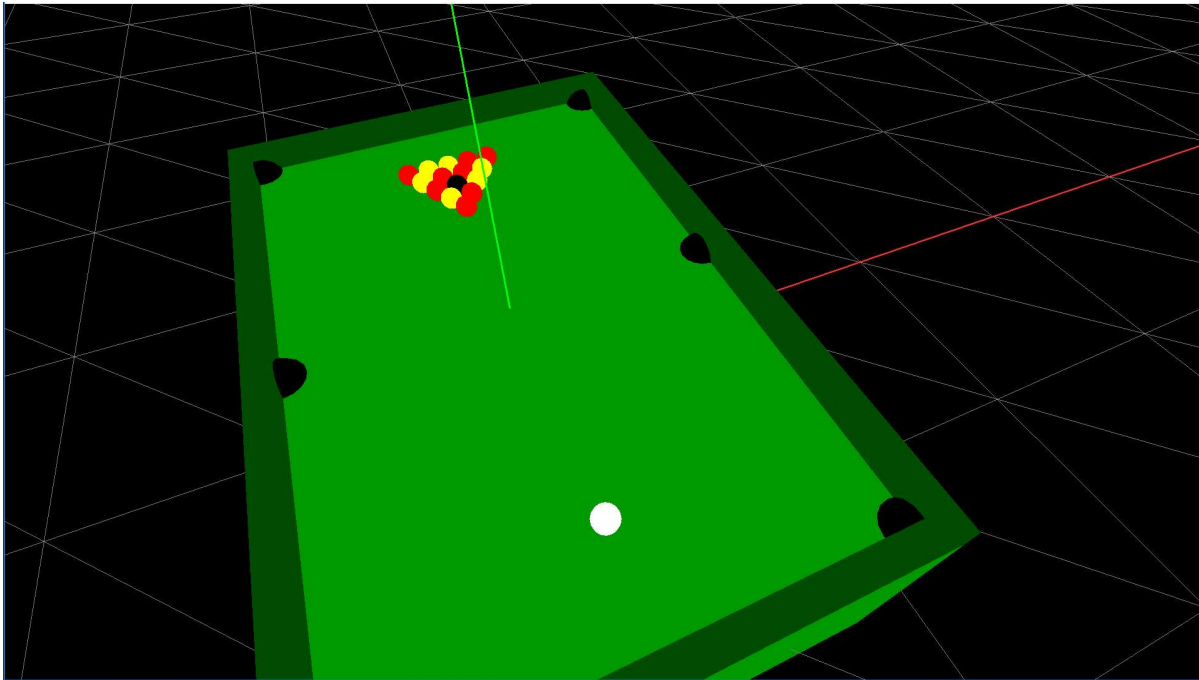
Joc de biliard

Scopul temei este de a crea un joc clasic de "8 Ball Pool" cu 2 jucatori, care foloseste 7 bile rosii, 7 bile galbene, o bila neagra si o bila alba. Exemplu: <https://www.youtube.com/watch?v=6uzf1nXZW6E> [https://www.youtube.com/watch?v=6uzf1nXZW6E]

Stadii

1. Spargerea bilelor

La inceputul jocului, bilele sunt aranjate ca in poza urmatoare:



Vor fi aranjate intr-un triunghi format din cele 15 bile in a treia treime a mesei, iar bila alba este situata in prima treime a mesei.

```

  R
 R G
G N R
G R R G
R R G G R

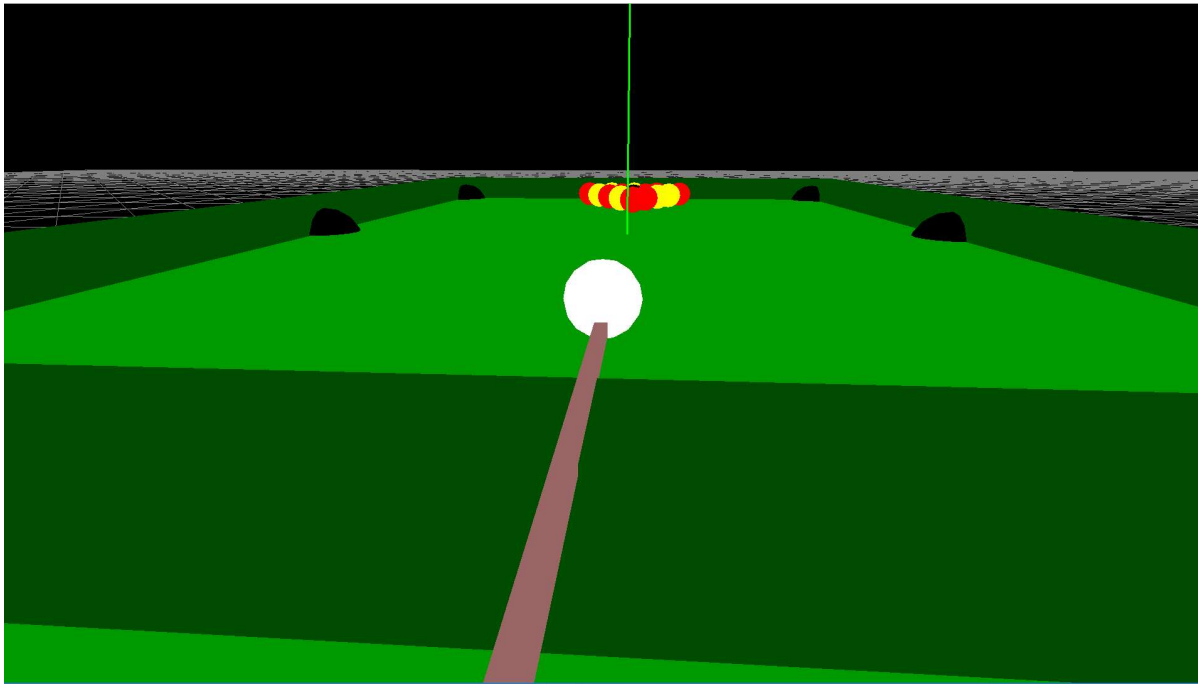
```

R - bila rosie G - bila galbena N - bila neagra

Jucatorul care incepe va putea deplasa bila alba in prima treime a mesei. Dupa ce a mutat bila unde doreste cu tastele "WASD", apasa pe "SPACE".

Jucatorul va fi constrans in a muta bila alba doar in prima treime a mesei

View-ul pe care il va avea dupa apasarea tastei "SPACE" este urmatorul:



La început, tacul este foarte aproape de bila alba. Tacul va avea o animație de retragere (pe direcția tacului) atunci când se apasă "Left Mouse Button". Această animație va fi făcută în **vertex shader**. După ce tacul a ajuns la o distanță anumită de bila alba, acesta revine aproape de bila alba și animația se reia. Intensitatea loviturii va fi direct proporțională cu distanța între tac și bila alba. Când dam drumul butonului din stanga al mouse-ului, se execută lovitura. De asemenea, cu "Right Mouse Button" se poate mișca tacul în jurul bilei albe, camera urmărind bila alba și tacul ca în modul third person. Direcția loviturii este dată de direcția tacului. După lovitura, bila alba va avea o anumită viteză cu direcția tacului și magnitudinea proporțională cu distanța dintre tac la bila alba în momentul release-ului butonului stâng al mouse-ului. În momentul ciocnirii bilei albe cu celelalte bile, acestea vor trebui să se comporte ca în viața reală. Deplasarea bilei albe este în planul XOZ.

2. Alegerea culorii fiecărui jucător

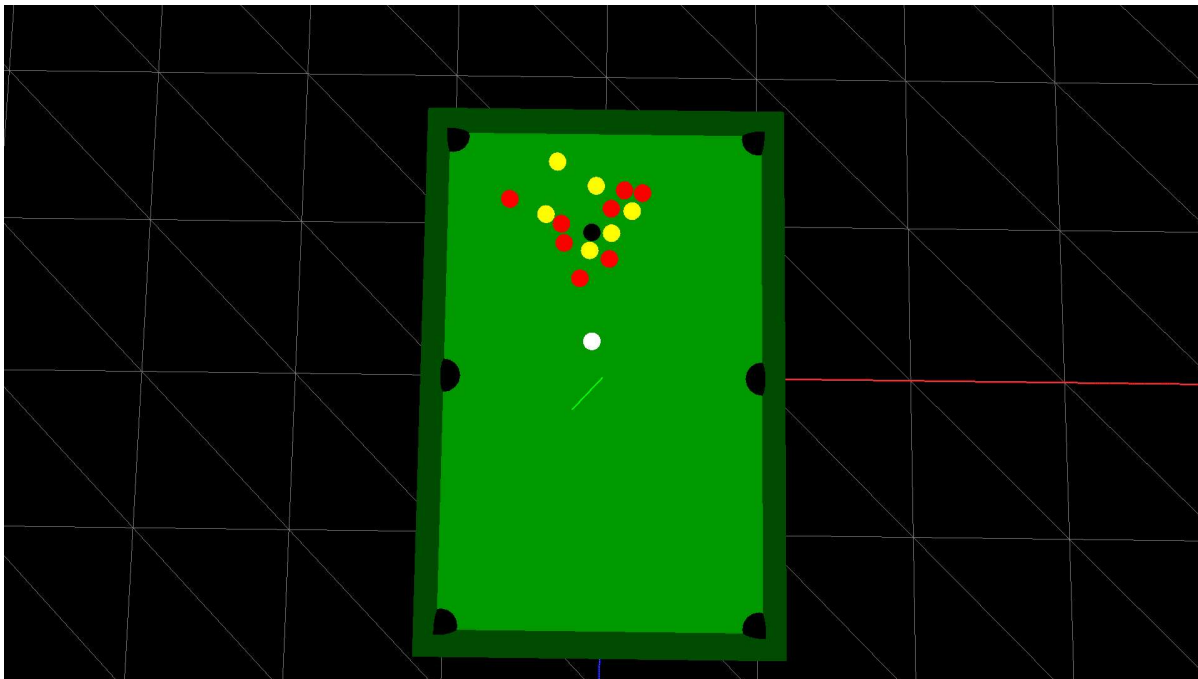
Culoarea cu care joacă un player este aleasă în momentul în care jucătorul introduce în buzunar prima bilă de culoare galbenă sau roșie. După aceea, jucătorul respectiv trebuie să introducă toate bilele de culoarea respectivă în buzunar, iar celălalt jucător trebuie să introducă bilele de cealaltă culoare.

3. Jocul până la bila neagră

Jucătorii trebuie să-și introducă în buzunar toate bilele. O bilă este introdusă în buzunar în momentul când se apropie atât de mult de un buzunar încât putem trage această concluzie. După aceea, ei pot introduce bila neagră. Cine introduce primul bila neagră este castigator. Dacă este comis un fault, celălalt jucător poate plasa bila alba oriunde pe masă cu tastele WASD (atâta timp cât nu intersectează alte bile). Fault-ul este de două feluri: dacă un jucător, la o lovitura, nu își atinge prima dată propriile bile sau dacă bila alba intră în buzunar. După ce a plasat bila alba pe masă cu un view top-down, apasă pe SPACE pentru a reveni la camera third person și jocul continuă.

Alte specificații

- După lovitura, camera se schimbă din third person pe bila alba pe una top-down, pentru a putea fi vizualizată lovitura:



- Pentru punctaj complet, jocul trebuie sa fie consecvent.
- Animatia tacului pentru lovirea bilei albe se va face in vertex shader.

Punctaj

- 10% - Afisare la stdout statistici joc (cate bile a bagat fiecare, cate fault-uri s-au comis etc)
- 10% - Implementarea coliziunilor (bila-bila, bila-manta) (link-ul acesta [http://www.gamasutra.com/view/feature/3015/pool_hall_lessons_fast_accurate_.php] ar fi de folos)
- 10% - Dispartia unei bile bagata in buzunar
- 15% - Animatie tac in vertex shader in momentul loviturii
- 15% - Camera third person pe directia tacului, target-ul fiind bila alba
- 15% - Vedere top-down dupa ce tacul a lovit bila ptr a se putea vizualiza lovitura
- 25% - Desenarea mesei cu buzunare si a bilelor, ca in poze

Bonus

Orice aduce realism jocului:

- Posibilitatea imprimarii unui efect lateral, in sus, in jos sau o combinatie intre ele (aici [<https://www.youtube.com/watch?v=0Skeucpa1zg>] si aici [<https://www.youtube.com/watch?v=kpRmoK3V6NQ>])
- Masa realista
- Linii pentru ghidare
- etc

Intrebari si raspunsuri

Pentru intrebari vom folosi forumurile de pe moodle.

Notare

Baremul este orientativ. Fiecare asistent are o anumita libertate in evaluarea temelor (de exemplu, sa dea punctaj partial pentru implementarea incompleta a unei functionalitati sau sa scada pentru hard coding). Acelasi lucru este valabil atat pentru functionalitatile obligatorii, cat si pentru bonusuri.

Tema trebuie incarcată pe moodle. Pentru a fi punctată, tema trebuie prezentată la laborator. Vor exista laboratoare speciale de prezentare a temelor (care vor fi anunțate).

Indicatii suplimentare

Tema va fi implementata in OpenGL si C++. Este indicat sa folositi framework-ul si Visual Studio.

Pentru implementarea temei, in folderul Source/Laboratoare/ puteti crea un nou folder, de exemplu Tema2, cu fisierele Tema2.cpp si Tema2.h (pentru implementare POO, este indicat sa aveti si alte fisiere). Pentru a vedea fisierele nou create in Visual Studio in Solution Explorer, apasati click dreapta pe filtrul Laboratoare si selectati Add→New Filter. Dupa ce creati un nou filtru, de exemplu Tema2, dati click dreapta si selectati Add→Existing Item. Astfel adaugati toate fisierele din folderul nou creat. In fisierul LabList.h trebuie adaugata si calea catre header-ul temei. De exemplu: #include <Laboratoare/Tema2/Tema2.h>

Arhivarea proiectului

- in mod normal arhiva trebuie sa contina toate resursele necesare compilarii si rularii
- inainte de a face arhiva asigurati-va ca ati dat clean la proiect
 - click dreapta pe proiect in **Solution Explorer** → **Clean Solution**, sau
 - stergeti folderul **/Visual Studio/obj**
- stergeti fisierul **/Visual Studio/Framework_EGC.sdf** (in caz ca exista)
- stergeti fisierul **/Visual Studio/Framework_EGC.VC.db** (in caz ca exista)
- stergeti folderul **/x64** sau **/x86** (in caz ca exista)
 - executabilul final este generat in folderul **/x86** sau **/x64** la finalul link-editarii in functie de arhitectura aleasa la compilare (32/64 biti)
- in cazul in care arhiva tot depaseste limita de 20MB (nu ar trebui), puteti sa stergeti si folderul **/libs** sau **/Resources** intrucat se pot adauga la testare. Nu este recomandat sa faceti acest lucru intrucat ingreuneaza mult testarea in cazul in care versiunea curenta a librariilor/resurselor difera de versiunea utilizata la momentul scrierii temei.

Deadline tema

5 decembrie 2018, ora 23:55

egc/teme/2018/02.txt · Last modified: 2018/12/04 17:42 by paul.gheorghe2411