

Simularea Cubului lui Rubik

Ioan Ștefan

Rezumat

Așa cum sugerează și titlul lucrării, obiectivul vizat a fost realizarea unei aplicații care să simuleze comportamentul Cubului lui Rubik $3 \times 3 \times 3$ (eng. Rubik's Cube), consacratul joc problemă* de tip puzzle inventat în 1974 de către sculptorul și profesorul de arhitectură maghiar Ernő Rubik. Pe un astfel de cub fiecare dintre cele șase fețe este acoperită cu nouă etichete, colorate într-una din culorile: alb, galben, portocaliu, roșu, albastru și verde (în mod tradițional). Un mecanism de pivoți permite rotirea independentă a fiecărei fețe astfel încât culorile se vor amesteca.

Pornind de la tema descrisă mai sus, scopul general l-a constituit realizarea unei **aplicații desktop care să simuleze Cubul lui Rubik într-un mediu virtual**. Implementarea care face obiectul aplicației în cauză trebuie să asigure, în primă fază, o *reprezentare grafică* cât mai fidelă a unui cub Rubik și, bineînțeles, *manipularea* acestuia într-o manieră care să faciliteze interactivitatea cu utilizatorul aplicației. În plus, s-a implementat o metodă de rezolvare automată a cubului Rubik, prin rotirea layerelor în ordine inversă și în sensuri opuse celor corespunzătoare amestecării cubului.

În acest sens, utilizând Visual C++ din cadrul mediului de dezvoltare software **Microsoft Visual Studio**, s-a creat o aplicație de tip Win32 console, iar codul aferent este scris în **limbajul de programare C++**.

Întrucât natura proiectului este de tip simulare grafică, a fost necesară utilizarea unui API (Application Programming Interface) specific. În acest sens, am ales **OpenGL (Open Graphics Library)**, o specificație a unui standard care definește un API procedural multiplatformă frecvent utilizat pentru programarea componentelor grafice 2D și 3D ale programelor de calculator (exemple de domenii de utilizare: grafică asistată de calculator, realitate virtuală, vizualizare științifică, simulări de zboruri, jocuri pe calculator). Funcția de bază a OpenGL este de a accepta primitive, cum ar fi puncte, linii și poligoane și de a le converti în pixeli. Acest lucru se face printr-un pipeline grafic, implementat în cadrul unei state-machine definit de OpenGL.

Majoritatea funcțiilor OpenGL folosite în program aparțin librăriei **GLUT (OpenGL Utility Toolkit)** prin intermediul căreia, printre altele, se poate defini și controla fereastra de lucru și se pot monitoriza input-urile de la tastatură și mouse.

* Jocul problemă este un joc ce pune la încercare logica, perspicacitatea, ingeniozitatea, perseverența și răbdarea celui ce încearcă să îl rezolve.