

## Adaugarea unui foc animat

Pentru foc vom folosi modelul: **fire.nfg**

De asemenea vom folosi texturile:

- **fire\_mask.tga** (atentie, trebuie inversata)
- **fire3.tga**
- **DisplacementMap.tga**

Modelul focului are o forma trapezoidala. Pentru a da o forma rotunjita si usor transparenta la marginea flacarilor folosim **fire\_mask** care va reprezenta componenta alpha a culorii focului.

Textura **DisplacementMap** e folosita pentru a realiza animatia flacarilor. Aceasta va fi deplasata pe verticala cu un anume pas.

Se va transmite catre fragment shader un float de tip uniform, sa ii spunem *u\_Time*, care va creste pe masura ce trece timpul de la inceperea aplicatiei (puteti folosi functia *clock()*). Atentie, pasul de crestere ar trebui sa fie mic (sa zicem la nivelul sutimilor, miimilor), pentru ca *u\_Time* trebuie sa se afle in intervalul  $[0,1)$ , iar cand ajunge mai mare sau egal cu 1 sa fie redus la partea fractionara (aceste calcule se fac in programul C++ nu in shader). Variabila *u\_Time* va fi folosita pentru a calcula coordonatele de displacement din textura *DisplacementMap*:

```
disp = texture2D(u_disp, vec2(v_uv.x, v_uv.y + u_Time)).rg;
```

Componentele r si g (echivalente cu x,y) vor reprezenta displacement-ul pe orizontala si respectiv verticala, a texturii.

Se va transmite si un uniform (tot float) *u\_DispMax*, care va da capetele intervalului de displacement:  $[-u\_DispMax, u\_DispMax]$ . Practic, *disp* trebuie raportat de la intervalul  $[0,1]$  la acest interval. Valoarea obtinuta va fi notata: *offset* si se va aduna la *v\_uv*, obtinand *v\_uv\_displaced*.

Se va prelua apoi culoarea din textura focului, insa cu coordonatele *v\_uv\_displaced*, si se va obtine *c\_fire*.

Pentru a pastra forma focului, in cazul lui **fire\_mask** se vor folosi coordonatele *v\_uv* initiale, obtinandu-se *c\_alpha*.

Apoi alpha-ul lui *c\_fire* se va inmulti cu oricare din componentele r,g sau b ale lui *c\_alpha* (fiind o nuanta de gri, componentele r,g,b sunt egale).

Culoarea obtinuta la final cu alpha-ul modificat va fi transmisa lui *gl\_FragColor*.

Nu uitati in fisierul cpp sa activati blendul:

```
glEnable (GL_BLEND);  
glBlendFunc (GL_SRC_ALPHA GL_ONE_MINUS_SRC_ALPHA);// pentru o aplicare  
corecta peste background (celelalte obiecte desenate)
```

**Obsevatie:** In fisierul de configurare, pentru foc, se va adauga un element cu valoarea pe care trebuie sa o ia *u\_DispMax*.