

Efecte vizuale

Mihai-Sorin Stupariu

Sem. al II-lea, 2022 - 2023

Principiul amestecării

- ▶ Legat de factorul $A(alpha, \alpha)$ din codul RGBA. Implicit $A = 1.0$ (obiectele sunt opace). Este specificat în cadrul funcțiilor de culoare / material. Elementele relevante:

Principiul amestecării

- ▶ Legat de factorul $A(alpha, \alpha)$ din codul RGBA. Implicit $A = 1.0$ (obiectele sunt opace). Este specificat în cadrul funcțiilor de culoare / material. Elementele relevante:
 - ▶ **Destinație** (fragment deja procesat)
 - ▶ Dat de $D = (R_d, G_d, B_d, A_d)$, $F_D = (D_r, D_g, D_b, D_a)$

Principiul amestecării

- ▶ Legat de factorul $A(alpha, \alpha)$ din codul RGBA. Implicit $A = 1.0$ (obiectele sunt opace). Este specificat în cadrul funcțiilor de culoare / material. Elementele relevante:
 - ▶ **Destinație** (fragment deja procesat)
 - ▶ Dat de $D = (R_d, G_d, B_d, A_d)$, $F_D = (D_r, D_g, D_b, D_a)$
 - ▶ **Sursa** (obiect procesat)
 - ▶ Dat de $S = (R_s, G_s, B_s, A_s)$, $F_S = (S_r, S_g, S_b, S_a)$

Principiul amestecării

- ▶ Legat de factorul $A(alpha, \alpha)$ din codul RGBA. Implicit $A = 1.0$ (obiectele sunt opace). Este specificat în cadrul funcțiilor de culoare / material. Elementele relevante:
 - ▶ **Destinație** (fragment deja procesat)
 - ▶ Dat de $D = (R_d, G_d, B_d, A_d)$, $F_D = (D_r, D_g, D_b, D_a)$
 - ▶ **Sursa** (obiect procesat)
 - ▶ Dat de $S = (R_s, G_s, B_s, A_s)$, $F_S = (S_r, S_g, S_b, S_a)$
- ▶ Factorul sursă F_S și factorul destinație F_D sunt indicați prin `glBlendFunc(srcfactor, destfactor)`

Principiul amestecării

- ▶ Legat de factorul $A(\alpha, \alpha)$ din codul RGBA. Implicit $A = 1.0$ (obiectele sunt opace). Este specificat în cadrul funcțiilor de culoare / material. Elementele relevante:
 - ▶ **Destinație** (fragment deja procesat)
 - ▶ Dat de $D = (R_d, G_d, B_d, A_d)$, $F_D = (D_r, D_g, D_b, D_a)$
 - ▶ **Sursa** (obiect procesat)
 - ▶ Dat de $S = (R_s, G_s, B_s, A_s)$, $F_S = (S_r, S_g, S_b, S_a)$
- ▶ Factorul sursă F_S și factorul destinație F_D sunt indicați prin `glBlendFunc(srcfactor, destfactor)`
- ▶ Factorul destinație (fragmentul deja procesat) și factorul sursă (obiectul care urmează să fie procesat și înregistrat) sunt "amestecate" utilizând o funcție $\varphi(D, F_d, S, F_s)$. Combinarea se realizează după formula

$$\varphi(D, F_d, S, F_s) = F_d * D + F_s * S, \quad (1)$$

urmată de 'clamp'.

Valori pentru factorii sursă / destinație

Constanta simbolică	Factor RGB	Factor A
GL_ZERO	$(0, 0, 0)$	0
GL_ONE	$(1, 1, 1)$	1
GL_SRC_ALPHA	(A_s, A_s, A_s)	A_s
GL_ONE_MINUS_SRC_ALPHA	$(1, 1, 1) - (A_s, A_s, A_s)$	$1 - A_s$
GL_DST_ALPHA	(A_d, A_d, A_d)	A_d
GL_ONE_MINUS_DST_ALPHA	$(1, 1, 1) - (A_d, A_d, A_d)$	$1 - A_d$
GL_SRC_COLOR	(R_s, G_s, B_s)	A_s
GL_ONE_MINUS_SRC_COLOR	$(1, 1, 1) - (R_s, G_s, B_s)$	$1 - A_s$
GL_DST_COLOR	(R_d, G_d, B_d)	A_d
GL_ONE_MINUS_DST_COLOR	$(1, 1, 1) - (R_d, G_d, B_d)$	$1 - A_d$
GL_CONSTANT_COLOR	(R_c, G_c, B_c)	A_c
GL_ONE_MINUS_CONSTANT_COLOR	$(1, 1, 1) - (R_c, G_c, B_c)$	$1 - A_c$
GL_CONSTANT_ALPHA	(A_c, A_c, A_c)	A_c
GL_ONE_MINUS_CONSTANT_ALPHA	$(1, 1, 1) - (A_c, A_c, A_c)$	$1 - A_c$
GL_SRC_ALPHA_SATURATE	$(f, f, f); f = \min(A_s, 1 - A_d)$	1

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz
Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatia_1$)

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz
 - Pas 1.** Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$(0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 =$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned}
 & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\
 & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\
 & = (0.5, 0.5, 0.0, 0.25)
 \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de GL_SRC_ALPHA și GL_ONE_MINUS_SRC_ALPHA (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.5, 0.5, 0.0, 0.25) \end{aligned}$$

Pas 2. Triunghiul turcoaz ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de GL_SRC_ALPHA și GL_ONE_MINUS_SRC_ALPHA (ambele valori sunt egale cu 0.5).

- **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatia_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.5, 0.5, 0.0, 0.25) \end{aligned}$$

Pas 2. Triunghiul turcoaz ($sursa_2$) pe ceea ce s-a desenat ($destinatia_2$):

$$(0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 =$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatia_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.5, 0.5, 0.0, 0.25) \end{aligned}$$

Pas 2. Triunghiul turcoaz ($sursa_2$) pe ceea ce s-a desenat ($destinatia_2$):

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.5, 0.5, 0.0, 0.25) = \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatia_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.5, 0.5, 0.0, 0.25) \end{aligned}$$

Pas 2. Triunghiul turcoaz ($sursa_2$) pe ceea ce s-a desenat ($destinatia_2$):

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.5, 0.5, 0.0, 0.25) = \\ & = (0.0, 0.5, 0.5, 0.25) + (0.25, 0.25, 0, 0.125) = \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (I)

- ▶ Sunt desenate un triunghi galben și un triunghi turcoaz (cyan) pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 1:** Ordinea: fundal negru, apoi triunghi galben, apoi triunghi turcoaz

Pas 1. Triunghiul galben ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.5, 0.5, 0.0, 0.25) \end{aligned}$$

Pas 2. Triunghiul turcoaz ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.5, 0.5, 0.0, 0.25) = \\ & = (0.0, 0.5, 0.5, 0.25) + (0.25, 0.25, 0, 0.125) = \\ & \quad (0.25, 0.75, 0.5, 0.375) \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben
Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben
 - Pas 1.** Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$(0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 =$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned}
 & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\
 & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\
 & = (0.0, 0.5, 0.5, 0.25)
 \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.0, 0.5, 0.5, 0.25) \end{aligned}$$

Pas 2. Triunghiul galben ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de GL_SRC_ALPHA și GL_ONE_MINUS_SRC_ALPHA (ambele valori sunt egale cu 0.5).

- **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.0, 0.5, 0.5, 0.25) \end{aligned}$$

Pas 2. Triunghiul galben ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

$$(0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 =$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.0, 0.5, 0.5, 0.25) \end{aligned}$$

Pas 2. Triunghiul galben ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.5, 0.5, 0.25) = \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).
- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned}
 & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\
 & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\
 & = (0.0, 0.5, 0.5, 0.25)
 \end{aligned}$$

Pas 2. Triunghiul galben ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

$$\begin{aligned}
 & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\
 & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.5, 0.5, 0.25) = \\
 & = (0.5, 0.5, 0.0, 0.25) + (0.0, 0.25, 0.25, 0.125) =
 \end{aligned}$$

Exemplu - Codul sursă 11_C_2_alpha2d.cpp (II)

- ▶ Sunt desenate un triunghi turcoaz (cyan) și un triunghi galben pe fundal negru, cu factor destinație și factor sursă dați de `GL_SRC_ALPHA` și `GL_ONE_MINUS_SRC_ALPHA` (ambele valori sunt egale cu 0.5).

- ▶ **Varianta 2:** Ordinea: fundal negru, apoi triunghi turcoaz, apoi triunghi galben

Pas 1. Triunghiul turcoaz ($sursa_1$) pe fundalul negru ($destinatie_1$)

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_1 + (0.5, 0.5, 0.5, 0.5) * destinatie_1 = \\ & = (0.5, 0.5, 0.5, 0.5) * (0.0, 1.0, 1.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.0, 0.0, 0.0) = \\ & = (0.0, 0.5, 0.5, 0.25) \end{aligned}$$

Pas 2. Triunghiul galben ($sursa_2$) pe ceea ce s-a desenat ($destinatie_2$):

$$\begin{aligned} & (0.5, 0.5, 0.5, 0.5) * sursa_2 + (0.5, 0.5, 0.5, 0.5) * destinatie_2 = \\ & = (0.5, 0.5, 0.5, 0.5) * (1.0, 1.0, 0.0, 0.5) + (0.5, 0.5, 0.5, 0.5) * (0.0, 0.5, 0.5, 0.25) = \\ & = (0.5, 0.5, 0.0, 0.25) + (0.0, 0.25, 0.25, 0.125) = \\ & \quad (0.5, 0.75, 0.25, 0.375) \neq (0.25, 0.75, 0.5, 0.375) \end{aligned}$$

Funcții de amestecare pentru scenele 3D

► Combine:

Funcții de amestecare pentru scenele 3D

- ▶ Combine:
 - ▶ ordinea în care sunt desenate obiectele
 - ▶ testul de adâncime
 - ▶ efectele de amestecare

Funcții de amestecare pentru scenele 3D

- ▶ Combinate:
 - ▶ ordinea în care sunt desenate obiectele
 - ▶ testul de adâncime
 - ▶ efectele de amestecare
- ▶ desenate obiecte opace cu:

Funcții de amestecare pentru scenele 3D

- ▶ Combine:
 - ▶ ordinea în care sunt desenate obiectele
 - ▶ testul de adâncime
 - ▶ efectele de amestecare
- ▶ desenate obiecte opace cu:
 - ▶ z-buffer activ
 - ▶ buffer de adâncime: normal (read/write) `glDepthMask(GL_TRUE)`
- ▶ desenate obiecte transparente cu:
 - ▶ z-buffer activ
 - ▶ buffer de adâncime: read `glDepthMask(GL_FALSE)`

Efectul de ceață

- Pentru a utiliza efectul de ceață trebuie să fie activate:

Efectul de ceață

- ▶ Pentru a utiliza efectul de ceață trebuie să fie activate:
 - ▶ amestecarea (**GL_BLEND**)

Efectul de ceață

- ▶ Pentru a utiliza efectul de ceață trebuie să fie activate:
 - ▶ amestecarea (`GL_BLEND`)
 - ▶ testul de adâncime (`GL_DEPTH_TEST`)

Efectul de ceață

- ▶ Pentru a utiliza efectul de ceață trebuie să fie activate:
 - ▶ amestecarea (`GL_BLEND`)
 - ▶ testul de adâncime (`GL_DEPTH_TEST`)
 - ▶ funcțiile de ceață (`GL_FOG`)

Efectul de ceață

- ▶ Pentru a utiliza efectul de ceață trebuie să fie activate:
 - ▶ amestecarea (**GL_BLEND**)
 - ▶ testul de adâncime (**GL_DEPTH_TEST**)
 - ▶ funcțiile de ceață (**GL_FOG**)
- ▶ **Principiu:** (mecanismul combinațiilor afine) este variată culoarea obiectelor în funcție de distanță, pe baza unei formule de tipul

$$C = f \cdot C_o + (1 - f) \cdot C_f,$$

unde: f = factor ceață; C_o = culoarea inițială a obiectului, C_f = culoarea ceții

Tipuri de ceață. Factorul ceață

- ▶ OpenGL are implementate trei tipuri de ceață: liniar, exponențial (implicit), exponențial pătratic

Tipuri de ceață. Factorul ceață

- ▶ OpenGL are implementate trei tipuri de ceață: liniar, exponențial (implicit), exponențial pătratic
- ▶ Factorul ceață f depinde de z -adâncime (\equiv depth) față de observator, fiind o funcție de forma $f = f(z)$, dată printr-o formulă de tipul

$$f(z) = \begin{cases} \frac{end-z}{z-start} & \text{(liniar)} \\ e^{-\rho z} & \text{(exponențial)} \\ e^{-\rho z^2} & \text{(exponențial pătratic)} \end{cases}$$

Tipuri de ceață. Factorul ceață

- ▶ OpenGL are implementate trei tipuri de ceață: liniar, exponențial (implicit), exponențial pătratic
- ▶ Factorul ceață f depinde de z -adâncime (\equiv depth) față de observator, fiind o funcție de forma $f = f(z)$, dată printr-o formulă de tipul

$$f(z) = \begin{cases} \frac{end-z}{z-start} & \text{(liniar)} \\ e^{-\rho z} & \text{(exponențial)} \\ e^{-\rho z^2} & \text{(exponențial pătratic)} \end{cases}$$

- ▶ Pentru tipul liniar sunt indicate valorile $start$, end (sunt implicit 0.0, respectiv 1.0)

Tipuri de ceață. Factorul ceață

- ▶ OpenGL are implementate trei tipuri de ceață: liniar, exponențial (implicit), exponențial pătratic
- ▶ Factorul ceață f depinde de z -adâncime (\equiv depth) față de observator, fiind o funcție de forma $f = f(z)$, dată printr-o formulă de tipul

$$f(z) = \begin{cases} \frac{end-z}{z-start} & \text{(liniar)} \\ e^{-\rho z} & \text{(exponențial)} \\ e^{-\rho z^2} & \text{(exponențial pătratic)} \end{cases}$$

- ▶ Pentru tipul liniar sunt indicate valorile $start$, end (sunt implicit 0.0, respectiv 1.0)
- ▶ Pentru tipul exponențial / exponențial pătratic este indicată ρ (densitatea ceții); implicit este 1.0.

Funcții asociate

- ▶ Funcția OpenGL specifică este

```
glFog* (parametru, valoare_parametru);
```

Funcții asociate

- Funcția OpenGL specifică este

`glFog* (parametru, valoare_parametru);`

- Parametrii considerați și valori asociate:

parametru	valoare_parametru
GL_FOG_MODE	GL_LINEAR
	GL_EXP (implicit)
	GL_EXP2
GL_FOG_DENSITY (ρ)	(implicit 1.0)
GL_FOG_START	(implicit 0.0)
GL_FOG_END	(implicit 1.0)
GL_FOG_COLOR	