

Normativa

1. Responen les següents dos preguntes en el mateix full de l'enunciat.
2. Cal que les respostes siguin **clares i justificades**.
3. No es poden usar apunts ni calculadores ni cap dispositiu electrònic.

1. (1 punt) Hem enviat a imprimir un dibuix de color vermell en una impressora CMY i en veure el dibuix imprès aquest es veu de color negre.

- a) Suposant que la impressora funciona correctament, de quin color és el paper en què s'ha imprès?

Solució:

Si la impressora funciona correctament el dibuix vermell (que en RGB és (1,0,0)) en format CMY és el (0,1,1) que vol dir que requereix de les tintes magenta i groga per a pintar-se. Si el dibuix es veu negre vol dir que aquest dibuix conté totes tres tintes CMY, i per tant això vol dir que necessàriament s'ha pintat en un paper de color cian. **El paper és cian.**

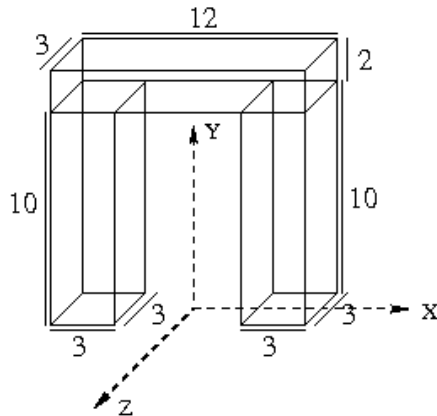
- b) Si el paper fos blanc i a la impressora li faltés la tinta cian de quin color veuríem el dibuix?

Solució:

El dibuix vermell no necessita la tinta cian perquè el vermell és composició de magenta i groc, per tant si es dibuixa en paper blanc el dibuix es veurà vermell i no li afecta que la impressora no tingui tinta cian. **El dibuix es veurà vermell.**

2. (2 punts) Disposem d'una funció `pintaCub()` que pinta un cub de costat 2 orientat amb els eixos i amb el seu vèrtex de coordenades mínimes a l'origen de coordenades. Es vol construir una escena que contingui una porta formada per dos columnes de base quadrada de costat 3 i alçada 10 i una biga sobre les dues columnes que medeix 12 d'amplada, 3 de profunditat i 2 d'alçada (veure imatge). Les bases de les dues columnes estan centrades als punts $(-4.5, 0, 0)$ i $(4.5, 0, 0)$ respectivament, i la base de la biga està centrada al punt $(0, 10, 0)$.

Indica quin seria el pseudocodi d'una funció `pintaEscena()` que permeti pintar l'escena descrita especificant el codi (o pseudocodi) que permet trobar les TGs requerides per a cada part de la porta i utilitzant el mètode `pintaCub()`. Justifica la resposta.



Solució:

La funció `pintaCub()` pinta un cub de costat 2 amb el vèrtex mínim al $(0,0,0)$ i el vèrtex màxim al $(2,2,2)$, per tant, el centre de la seua base es troba al punt $(1,0,1)$.

Per a construir la columna de l'esquerra, el primer que farem és portar el centre de la base del cub a l'origen (Translació $(-1,0,-1)$), després l'hem d'escalar per a què faci 3 en X i Z i 10 en Y, per tant farem un escalat no uniforme amb valors d'escala de $3/2$ en X i Z i $10/2$ en Y (Escalat $(1.5,5,1.5)$), i finalment portarem el centre de la base d'aquest objecte al punt desitjat (Translació $(-4.5,0,0)$). És a dir la TG d'aquesta columna serà: $TG1 = Translacio(-4.5,0,0) * Escalat(1.5,5,1.5) * Translacio(-1,0,-1)$

En el cas de la columna de la dreta, farem exactament el mateix excepte per l'última translació al punt final, que en aquest cas serà Translació $(4.5,0,0)$, per tant la TG serà: $TG2 = Translacio(4.5,0,0) * Escalat(1.5,5,1.5) * Translacio(-1,0,-1)$

Finalment la biga comencem igualment portant el centre de la base del cub a l'origen però després l'escalat en aquest cas és diferent, perquè necessitem un factor d'escala de $12/2$ per a X, no cal escalar en Y i un factor de $3/2$ en Z (Escalat $(6,1,1.5)$). Finalment el centre de la base de l'objecte l'hem de translladar al punt desitjat (Translació $(0,10,0)$). És a dir la TG serà: $TG3 = Translacio(0,10,0) * Escalat(6,1,1.5) * Translacio(-1,0,-1)$

El pseudocodi de la funció `pintaEscena()` serà:

```
pintaEscena() {
    TG1 = Translació (-4.5,0,0);
    TG1 = TG1 * Escalat (1.5,5,1.5);
    TG1 = TG1 * Translació (-1,0,-1);
    modelMatrix (TG1);
    pintaCub();
    TG2 = Translació (4.5,0,0);
    TG2 = TG2 * Escalat (1.5,5,1.5);
    TG2 = TG2 * Translació (-1,0,-1);
    modelMatrix (TG2);
    pintaCub();

    TG3 = Translació (0,10,0);
    TG3 = TG3 * Escalat (6,1,1.5);
    TG3 = TG3 * Translació (-1,0,-1);
    modelMatrix (TG3);
    pintaCub();
}
```

Nom i cognoms:

Normativa del test

- (a) A les graelles que hi ha a continuació, marca amb una creu les teves respostes de l'examen. **No es tindrà en compte cap resposta fora d'aquestes graelles.**
- (b) No es poden usar apunts, calculadores ni cap dispositiu electrònic.
- (c) Totes les preguntes tenen una única resposta correcta.
- (d) Les preguntes contestades de forma errònia tenen una **penalització del 33%**.

Num	A	B	C	D
3	X			
4		X		
5	X			
6		X		
7			X	

Num	A	B	C	D
8			X	
9				X
10			X	
11	X			
12			X	

3. (1 punt) Quins dels següents paràmetres de posició i orientació de la càmera permetrien veure l'escena de l'exercici 2 en forma de lletra C?
- OBS = (0,6,6); VRP = (0,6,0); up = (1,0,0).
 - OBS = (0,6,5); VRP = (0,6,0); up = (0,1,0).
 - OBS = (0,6,-5); VRP = (0,0,0); up = (1,0,0).
 - Cap de les altres respostes és correcta.
4. (1 punt) Suposem una càmera amb paràmetres OBS=(10,6,0); VRP=(0,6,0); i up=(0,0,-1) per a veure l'escena de l'exercici 2. Quina de les següents transformacions geomètriques aconseguiria la mateixa View Matrix (VM) que aconseguiria la crida lookAt(OBS,VRP,up)?
- $VM = T(0,0,-10) * R_z(90) * R_x(90) * T(0,-6,0)$
 - $VM = T(0,0,-10) * R_z(90) * R_y(-90) * T(0,-6,0)$
 - $VM = T(0,-6,0) * R_y(-90) * R_z(90) * T(0,0,-10)$
 - $VM = T(0,-6,0) * R_y(-90) * T(0,0,-10)$
5. (1 punt) Tenim un cub de costat 2 amb vèrtex de coordenades mínimes a l'origen de coordenades. Quina de les següents seqüències d'inicialització de posició de càmera permetrien (suposant que l'òptica és ortogonal i és correcta) veure en pantalla un hexàgon? (Nota: Considera els angles en graus)
- OBS=(3,3,3); VRP=(0,0,0); up=(0,1,0);
VM = lookAt (OBS, VRP, up);
viewMatrix(VM);
 - OBS=(2,2,2); VRP=(0,0,0); up=(1,1,1);
VM = lookAt (OBS, VRP, up);
viewMatrix(VM);
 - VM = Translate (0,0,-7);
VM = VM * Rotate_Y (45);
VM = VM * Translate (-1,-1,-1);
viewMatrix(VM);
 - VM = Translate (0,0,-3);
VM = VM * Rotate_Z (45);
VM = VM * Rotate_Y (-45);
VM = VM * Translate (-1,-1,-1);
viewMatrix(VM);

6. (1 punt) Donada l'escena del cub de l'exercici 5 i amb una càmera ortogonal que permet veure en pantalla un hexàgon sencer centrat al viewport, quin efecte tindria en la imatge si a aquesta càmera li anem modificant el vector d'up i actualitzant la matriu de VM corresponentment?

- a) Haurem de modificar la raw per continuar veient tot l'hexàgon sense retallar.
- b) Es veurà l'hexàgon igual que es veia però girat respecte del seu centre.
- c) No ho podem saber si no coneixem exactament l'òptica de la càmera.
- d) En comptes d'un hexàgon veurem un rectangle.

7. (0.5 punts) Indica quina de les següents associacions relaciona correctament cada matriu de transformació de Sistema de Coordenades amb el canvi de Sistemes de Coordenades que realitza. Recorda que els sistemes de coordenades són: SCM: Sistema de Coordenades de Model; SCA: Sistema de Coordenades d'Aplicació; SCO: Sistema de Coordenades d'Observador; SCC: Sistema de Coordenades de Clipping.

- a) VM: $SCM \rightarrow SCO$; PM: $SCO \rightarrow SCC$; TG: $SCA \rightarrow SCM$
- b) PM: $SCO \rightarrow SCC$; TG: $SCM \rightarrow SCO$; VM: $SCA \rightarrow SCM$
- c) PM: $SCO \rightarrow SCC$; TG: $SCM \rightarrow SCA$; VM: $SCA \rightarrow SCO$
- d) PM: $SCA \rightarrow SCO$; TG: $SCM \rightarrow SCA$; VM: $SCO \rightarrow SCC$

8. (0,5 punts) Donat el següent codi en el Vertex Shader, i on VM és la view matrix, podem afirmar que:

```
gl_Position = VM * vec4 (vertex,1);
```

- a) Que el punt (0,0,0) de SCA està dins del Volum de Visió
- b) Que el retallat no funcionarà perquè no estem multiplicant el vèrtex per cap matriu de projecció.
- c) Que el volum de visió és un cub que en SCO va entre (-1,-1,-1) i (1,1,1)
- d) Que el volum de visió és un cub que en SCA va entre (-1,-1,-1) i (1,1,1)

9. (0.5 punts) Per a què el procés de visualització funcioni correctament, cal que:

- a) En el Vertex Shader calculem les coordenades de clipping de cada vèrtex i en el Fragment Shader calculem les coordenades de dispositiu i el color.
- b) En el Vertex Shader calculem les coordenades de dispositiu de cada vèrtex i en el Fragment Shader assignem color als fragments.
- c) Com a sortida del Vertex Shader tinguem el color del vèrtex i en el Fragment Shader calculem les coordenades del fragment.
- d) Cap de les altres respostes és correcta.

10. (0.5 punts) Donada una view Matrix (VM) i un Viewport. Què cal assegurar en els paràmetres de la project Matrix (PM) per a evitar que l'escena que es pinta quedi deformada?
- a) Si $rav < 1$ cal igualar raw amb rav ($raw = rav$), sinó no.
 - b) Només cal augmentar el fov.
 - c) Només cal igualar raw amb rav ($raw = rav$).
 - d) No hi ha prou amb igualar raw amb rav ($raw = rav$), cal també modificar el fov.
11. (0.5 punts) Es vol definir una càmera perspectiva per veure una escena en tercera persona. Hem ubicat VRP al centre de l'esfera contenidora de l'escena. L'esfera té radi R. L'observador està fora de l'esfera a una distància d del VRP. Sabem que el viewport és quadrat. Per calcular el FOV d'una càmera perspectiva que permeti veure l'esfera sencera digues quina afirmació és certa:
- a) Ho podem fer independentment del valor de Znear.
 - b) Cal conèixer Znear i Zfar.
 - c) Si coneixem Znear no ens cal el valor de d.
 - d) Cap de les altres respostes és correcta.
12. (0.5 punts) Què es pot dir d'un color que en HSB (o HSV) té un valor màxim en la component S?
- a) Que és un color gris i la seva intensitat ve donada pel valor de B (o V).
 - b) Que és un color pur i per tant com a mínim un dels components del model RGB ha de ser 1.
 - c) Que és un color pur i per tant com a mínim un dels components del model RGB ha de ser 0.
 - d) No en podem dir res sense saber els valors de H i de B (o V).