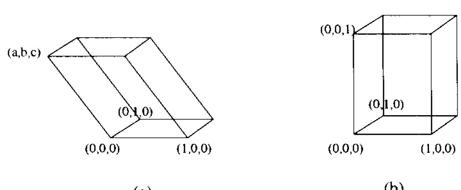
TENTAMEN COMPUTER GRAPHICS EN COMPUTATIONAL GEOMETRY

23 augustus 2002, 9:00 uur

Het tentamen bestaat uit de onderstaande **vier** *opgaven. Bij elk van de opgaven is het maximale aantal voor deze opgave te behalen punten vermeld. Je krijgt 10 punten gratis.*

Opgave 1 *(25 pt.)*
Het Cohen-Sutherland algoritme clipt een lijnstuk AB, met beginpunt $A = (x_1, y_1)$ en eindpunt $B = (x_2, y_2)$, t.o.v. een rechthoekig window W in het vlak. Dit gebeurt aan de hand van gebiedscodes die de relatieve posities van de eindpunten van het lijnstuk t.o.v. de windowranden aangeven. Beschouw nu *drie-dimensionale* lijn clipping.
a. Neem in eerste instantie aan dat het kijkvolume een *kubus* is. Het Cohen-Sutherland algoritme kan worden aangepast voor lijn clipping t.o.v. deze kubus W. Hoe moeten nu de verschillende gebieden van de 3D ruimte gecodeerd worden? Hoe kan uit de codes van de eindpunten, code(A) en code(B), bepaald worden of AB geheel binnen, dan wel geheel buiten, de kubus W valt?
b. In het geval van scheve parallelprojectie dient het kijkvolume eerst te worden getransformeerd naar een genormaliseerd kijkvolume. Het standaard kijkvolume voor *loodrechte* parallelprojectie is de kubus W begrensd door de vlakken $x = 0, x = 1, y = 0, y = 1, z = 0, z = 1$, met front-plane $z = 0$ en back-plane $z = 1$, zie Fig. 1(b). Dit komt overeen met de verzameling punten van de vorm $\vec{r} = \lambda_1\vec{e}_1 + \lambda_2\vec{e}_2 + \lambda_3\vec{e}_3$ met $0 \le \lambda_i \le 1$, waarbij de vectoren $\vec{e}_1, \vec{e}_2, \vec{e}_3$ gegeven worden door $\vec{e}_1 = (1,0,0), \vec{e}_2 = (0,1,0), \vec{e}_3 = (0,0,1)$. Het standaard kijkvolume voor *scheve* parallelprojectie in de richting gegeven door de richtingsvector (a,b,c) is het gedeelte van de ruimte opgespannen zoals boven, waarbij nu de vectoren worden gegeven door $\vec{v}_1 = (1,0,0), \vec{v}_2 = (0,1,0), \vec{v}_3 = (a,b,c)$, met front-plane $z = 0$ en back-plane $z = c$, zie Fig. 1(a).



Figuur 1: Kijkvolume voor (a) scheve, en (b) loodrechte parallelprojectie.

We willen nu het kijkvolume voor scheve projectie transformeren naar het kijkvolume voor loodrechte projectie, middels een lineaire transformatie $\mathcal{M} : (x, y, z) \to (x', y', z')$.
Leidt de 3×3 matrix M behorende bij de transformatie $\mathcal{M}$ af. (*Aanwijzing*: $\mathcal{M}$ beeldt de vectoren $\vec{v}_1, \vec{v}_2, \vec{v}_3$ af op resp. $\vec{e}_1, \vec{e}_2, \vec{e}_3$. Dit legt de matrix M vast.) Geef ook de 4×4 matrix van de transformatie $\mathcal{M}$ in homogene coordinaten.

Opgave 2 *(20 pt.)*
Beschouw de volgende methoden voor shading van polygonale oppervlakken: (i) Flat shading (ii) Gouraud shading (iii) Phong shading.
a. Beschrijf in het kort elk van deze methoden, en geef de onderlinge verschillen aan.
b. Beschouw nu shading van een een driehoek. Neem aan dat alle drie vertexnormalen van deze driehoek gelijk zijn aan de normaal van het vlak waarin de driehoek ligt. Worden in dit geval positie en/of intensiteit van het highlight bij spiegelreflectie altijd correct weergegeven bij toepassing van (i) Gouraud shading (ii) Phong shading ? Motiveer je antwoord.

Opgave 3 *(20 pt.)*
Gegeven is een verzameling L bestaande uit n niet-verticale lijnen in het platte vlak. De onder-baai van L is de verzameling punten in het platte vlak die niet boven een lijn uit L liggen (een punt uit de onder-baai mag dus wel *op* één of meer lijnen uit L liggen).

1. Bewijs dat de onder-baai van L een convexe verzameling is.

Z. 0-2

2. Beschrijf een algoritme dat in $O(n)$ *randomized expected time* en $O(n)$ *worst-case* geheugen het punt van de onder-baai berekent met maximale y-coördinaat. Voor de afleiding van de tijds-complexiteit mag je verwijzen naar de theorie.

Opgave 4 *(25 pt.)*
Gegeven is een verzameling S bestaande uit n niet-verticale lijnsegmenten in het platte vlak. Geen enkel tweetal randpunten van segmenten uit S ligt op een verticale lijn.

1. Geef een algoritme dat in $O(n)$ tijd een rechthoek bepaalt met horizontale en verticale zijden die alle segmenten uit S bevat.

2. Geef de definitie van *trapezoidal map* van de verzameling S.

3. Geef een deterministisch algoritme dat in $O(n \log n)$ tijd en $O(n)$ geheugen de trapezoidal map van S construeert. (Een algoritme heet deterministisch als het geen random keuzen maakt.)