

# Euler's angles

## Objectiu:

- L'objectiu d'aquest exercici és mostrar un objecte (definit al fitxer a p1\_escena.pov mitjançant el llenguatge de descripció d'escena de Pov-Ray) que mostra una seqüència contínua en què gira des de la seva orientació original a  $t = 0$  fins a una nova orientació a  $t = 1$  especificat mitjançant la seva matriu de rotació  $3 \times 3$ .
- Les dades completes de la matriu de rotació  $3 \times 3$  s'obtindran de les vostres dades personals codificades al número de document d'identificació, tal com s'explica a continuació.

## Procediment:

La tasca a realitzar es dividirà en els punts següents:

- (a) Construïu la matriu de rotació completa, amb tots els seus 9 elements reals diferents de zero, escollint 3 elements, assignant-los valors numèrics a partir de les dades del número del vostre document d'identificació i derivant els elements restants de les propietats generals d'una rotació. matriu.
  - Per exemple, si denotem els elements de la primera columna per  $[x1, x2, x3]$ ; i  $[y1, y2, y3]$ ,  $[z1, z2, z3]$  per a la segona i tercera columna respectivament, estan subjectes a les relacions:
    - $[x1, x2, x3] \cdot [x1, x2, x3] = 1$
    - $[y1, y2, y3] \cdot [y1, y2, y3] = 1$
    - $[z1, z2, z3] \cdot [z1, z2, z3] = 1$
  - ,
    - $[x1, x2, x3] \cdot [y1, y2, y3] = 0$
    - $[x1, x2, x3] \cdot [z1, z2, z3] = 0$
    - $[y1, y2, y3] \cdot [z1, z2, z3] = 0$
  - ,
  - ,
  - I també condicions similars per a les files:
    - $[x1, y1, z1] \cdot [x1, y1, z1] = 1$
    - $[x2, y2, z2] \cdot [x2, y2, z2] = 1$
    - $[x3, y3, z3] \cdot [x3, y3, z3] = 1$
  - ,
    - $[x1, y1, z1] \cdot [x2, y2, z2] = 0$
    - $[x1, y1, z1] \cdot [x3, y3, z3] = 0$
    - $[x2, y2, z2] \cdot [x3, y3, z3] = 0$
  - ,
  - ,
  - ,
  - Per tant, podeu escollir a voluntat 3 dels 9 elements, per exemple, els valors  $x1, x3, z1$ , assigneu-los valors en l'interval  $[0,1]$  prenent 2 dígit successius del vostre identificador. número invertit i, a continuació, trobeu els valors restants fent ús de les expressions anteriors, una per una. Per exemple, si el vostre número d'identificació és
  - DNI # = 41489762, assignareu els valors 0.26, 0.79, 0.84 als elements escollits. Us recomanem que escolliu les posicions dels 3 elements de manera que exactament 2 d'ells pertanyin a la mateixa columna, i el tercer pertanyi, òbviament, a una columna diferent. La raó és que si en una columna (o fila per a això) coneixem 2 elements, el tercer element es troba immediatament pel fet que tres elements formen un vector unitari ( $e1^2 + e2^2 + e3^2 = 1$ ).
  - Així, per exemple, a partir dels valors de  $x1, x3, z1$  podeu trobar immediatament els valors adequats per a  $x2$  a la columna  $[x1, x2, x3]$  i  $y1$  a la fila  $[x1, y1, z1]$ .
  - Finalment, una vegada que es coneix completament una columna, si en una columna diferent també es coneix un element, el fet que les dues columnes siguin ortonormals permet una reconstrucció simple dels valors desconeguts. Fer-ho per a cada columna permetrà reconstruir completament la matriu de rotació de només 3 elements i, per tant, complir el punt a). (Si teniu dificultats, primer podeu continuar amb els punts restants utilitzant primer qualsevol de les matrius de rotació disponibles al subdirectori final).
  - A més, es proporcionen les dades de rotació de matriu següents per a proves:
    - 0,53632 -0,30202 0,78813 1a fila
    - 0,68301 0,70388 -0,19506 2a fila
    - -0,49583 0,64292 0,58379 3a fila
  - Tingueu en compte que els elements de la matriu de rotació de la prova s'emmagatzemen al fitxer rijxcolumnes (observeu que l'ordre són primer elements de columna, després elements de segona columna i després tercers).
  - ,
  - ,
  - Heu de començar a visualitzar l'escena, executant les ordres:
    - `povray +w640 +h480 +a0.2 p1_escena.pov`
  - que crearà el fitxer p1\_escena.png que podeu visualitzar fent-hi clic o mitjançant la línia d'ordres, per exemple escrivint:
    - `eog p1_escena.png` or
    - `xv p1_escena.png`
  - Fixeu-vos que, associat a l'objecte i unit a aquest, hi ha un sistema de referència amb eix en colors vermell, verd i blau.

- (b) Haureu d'esbrinar quines 3 rotacions elementals consecutives R1 (z,  $f_{i_z}$ ) R2 (y,  $f_{i_y}$ ) R3 (x,  $f_{i_x}$ ) aconseguirà el canvi desitjat en l'orientació de l'objecte i crear una imatge animada. seqüència de fotogrames que va des de l'orientació inicial ( $f_{i_i} = 0$ ) a  $t = 0$  fins als valors calculats  $f_{i_z}$ ,  $f_{i_y}$ ,  $f_{i_x}$  a  $t = 1$ .

## Complete the following steps:

1. Esbrineu la matriu de rotació general (simbòlica) 3x3 expressada en termes de les 3 variables  $f_{i_i}$  (angles d'Euler) que entren a R1 (z,  $f_{i_z}$ ) R2 (y,  $f_{i_y}$ ) R3 (x,  $f_{i_x}$ ) (només es fa el producte de ho fan les 3 matrius).
2. Escriviu les equacions que relacionen els valors  $r_{ij}$  de la matriu general 3x3 amb  $f_{i_z}$ ,  $f_{i_y}$ ,  $f_{i_x}$  angles.
3. Escriviu un algorisme que permeti trobar  $f_{i_z}$ ,  $f_{i_y}$ ,  $f_{i_x}$  a partir dels valors  $r_{ij}$  (vegeu el punt 5 més avall)
4. Implementeu-lo en un programa que llegirà el fitxer rijxcolumnes i calcularà i imprimirà els valors de  $f_{i_z}$ ,  $f_{i_y}$ ,  $f_{i_x}$  al fitxer fisef.out (separats per comes).
5. executeu el programa, generant el fitxer fisef.out-
  - (Tingueu en compte que aquest fitxer ja existeix, de manera que podeu comprovar si els vostres programes funcionen correctament). Si és així, podreu completar els punts següents.
6. Feu una ullada al fitxer p1\_verifica.pov, especialment a les línies:
  - #fopen Rij "rijxcolumnes" read
  - #read (Rij,r11,r21,r31,r12,r22,r32,r13,r23,r33)
  - #fopen Fis "fisef.out" read
  - #read (Fis,f1z,f1y,f1x)
  - ,
  - ,
  - El codi Pov-Ray llegeix tant els elements  $r_{ij}$  com els angles d'Euler (Fis, f1z, f1y, f1x). Tingueu en compte que l'orientació desitjada provinent de  $r_{ij}$  es mostra en forma de sistema de referència en color gris (objecte SRob, Sistema de Referència objectiu):
    - // Escena
    - object {SRob matrix < r11,r21,r31,r12,r22,r32,r13,r23,r33, 0.0, 0.0, 0.0 > }
    - //
  - i, d'altra banda, la representació de l'objecte (i el sistema de referència de colors que s'hi fixa) es basa en els angles d'Euler:
    - object{Roda\_dentada rotate f1x\*x rotate f1y\*y rotate f1z\*z}
    - object {SRef rotate f1x\*x rotate f1y\*y rotate f1z\*z}
  - Tingueu en compte que la primera transformació aplicada a les coordenades dels punts objecte és R (x), la segona R (y) i l'última R (z), és a dir, l'ordre invers de la que s'utilitza al punt 1. Això és degut a al fet que Pov-Ray utilitza sempre el sistema de referència universal a l'hora d'aplicar transformacions.
  - Representar l'escena:
    - povray +w800 +h600 +a0.2 p1\_verifica.pov
    - eom p1\_verifica.png
  - Si tot va bé, els 3 eixos grisos i els de color coincidiran (és a dir, l'orientació generada per les 3 rotacions elementals que heu calculat és correcta, és a dir, reproduïxen els 9 elements complets de la matriu de rotació necessària). Si aquest és el cas, els eixos grisos no seran visibles, ja que es troben dins dels de colors, de diàmetre lleugerament més gran.
7. creeu un directori nou, on generareu una seqüència d'imatges que mostren una orientació dependent del temps que varia contínuament. Copieu el fitxer rijxcolumnes i substituïu els valors pel de la matriu que heu construït al punt a). (Si no ho feu, podeu fer servir qualsevol de les matrius de rotació disponibles al subdirectori final). Copieu també els diversos fitxers .pov i .ini i torneu a executar el programa, generant el fitxer fisef.out per a la vostra matriu. Es requereix que la seqüència satisfaci que a  $t = 0$  l'orientació ve donada per la matriu d'identitat (és a dir, la que mostra p1\_escena.pov) mentre que a  $t = 1$  ha de coincidir amb l'orientació final que mostra p1\_verifica.pov.
  - Pov-Ray utilitza una variable anomenada rellotge els valors dels quals s'especifiquen en un fitxer d'inicialització (p1\_anim.ini en aquest exemple), de manera que un sol codi povray pot generar una seqüència d'imatges variable.
  - Per aconseguir-ho, substituïrem la línia
    - rotate f1x\*x rotate f1y\*y rotate f1z\*z
    - by
    - rotate Factor\*f1x\*x rotate Factor\*f1y\*y rotate Factor\*f1z\*z
  - on Factor és un valor dependent del rellotge que canvia de 0 a 1 a mesura que passa el temps. (Això ja s'ha fet als fitxers p1\_anim.pov i p1\_anim.ini, i només heu de modificar els paràmetres de p1\_anim.ini relacionats amb el nombre de fotogrames inclosos en l'interval entre  $t = 0$  i  $t = 1$ ).
  - L'efecte del valor de temps (valor real de la variable de rellotge) sobre la imatge representada es pot examinar d'una manera senzilla mitjançant l'opció de línia d'ordres + K. Per exemple,
    - povray +w800 +h600 +a0.2 +K0.2 p1\_anim.pov
  - generarà l'escena amb un valor real de la variable de rellotge igual a 0,2
  - Quan estigueu segurs que diversos valors de rellotge de l'interval [0-1] proporcionen resultats raonables, podeu executar-los
    - povray +w800 +h600 +a0.2 p1\_anim. ini
8. Feu una ullada a la quantitat d'imatges PNG. Si emeteu l'ordre
  - eom \*png
  - podeu passar imatges ràpidament cap endavant prement la tecla espaiadora i cap enrere prement la tecla de retrocés. Genereu un fitxer mpg amb l'ordre convert.

9. A l'animació acabada de crear, vam fer que els 3 angles d'Euler canviessin linealment amb el temps, ja que el valor del rellotge creix uniformement entre els valors definits a p1\_anim.ini i les línies

- `// #declare Factor=sin(pi*clock)*sin(pi*clock);`
- `#declare Factor=clock;`
- fer que el factor sigui trivialment igual al rellotge. Creeu una nova animació en què els angles de rotació  $\text{Factor} * f1x$ ,  $\text{Factor} * f1y$ ,  $\text{Factor} * f1z$  segueixin una dependència temporal del tipus  $\sin^2(t)$ , simplement comentant i descomentant les línies adequades que declaren la variable Factor al fitxer p1\_anim.pov.

10. Finalment, repetiu els punts anteriors utilitzant la mateixa matriu de rotació que prové de les vostres dades id # amb una altra caracterització d'angles d'Euler, diferent de l'ordenació convencional R1 (z, fi\_z) R2 (y, fi\_y) R3 (x, fi\_x), és a dir, escollint un altre conjunt de tres rotacions elementals (segons el vostre criteri). Haureu de calcular el contingut simbòlic de la matriu de rotació general associada a l'ordre escollit, crear un codi ew que calculi els 3 nous valors d'angle d'Euler, generi el fitxer fisef.out i modifiqueu les línies de p1\_anim.pov:

- `#fopen Fis "fisef.out" read`
- `#read (Fis,f1z,f1y,f1x)`
- ,
- i
- ,
- `rotate f1x*x rotate f1y*y rotate f1z*z`
- and/or
- `rotate Factor*f1x*x rotate Factor*f1y*y rotate Factor*f1z*z`
- ,
- de manera que reflecteixin en conseqüència l'ordre que heu escollit per parametritzar la matriu de rotació en termes d'angles d'Euler.

Entregueu un informe que descrigui la tasca, els problemes trobats i les solucions, el vostre codi per calcular els angles d'Euler en les dues versions d'ordres, els canvis introduïts als fitxers .pov i les seqüències animades.