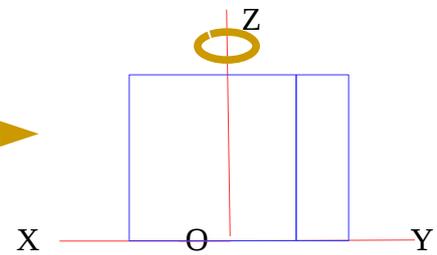
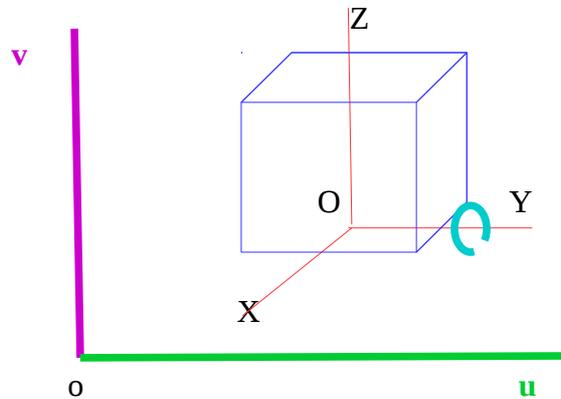


(1) On effectue une rotation d'angle Psi autour de OZ →



Vue de profil, vous avez l'axe OX dans l'oeil !

(2) On effectue une rotation d'angle Fi autour de OY →



Pour convertir les coordonnées 3D (x, y, z) et « plaquer » le solide dans le plan repéré, on calcule les coordonnées 2D (u, v) :

```

function abs_ouv(x, y, z : real) : word; // u:= abs_ouv(x, y, z)
begin abs_ouv:=trunc(x*cos(Fi)+y*sin(Fi)); end;
{=====}
function ord_ouv(x, y, z : real) : word; // v:=ord_ouv(x, y, z)
begin ord_ouv:=trunc(z*cos(Psi)+y*sin(Psi)); end;

```

Programmation de la rotation en 3D :

A un point de coordonnées (x0, y0, z0), on applique la rotation autour de l'axe OZ, d'un angle teta. Les coordonnées (x1, y1, z1) du point image sont calculées ainsi :

$$\begin{vmatrix} x_0 \\ y_0 \\ z_0 \end{vmatrix} \times \begin{vmatrix} \cos(\text{teta}) & -\sin(\text{teta}) & 0 \\ \sin(\text{teta}) & \cos(\text{teta}) & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x_1 \\ y_1 \\ z_1 \end{vmatrix}$$

Matrice de rotation
d'angle teta autour
de l'axe OZ

```

function abs_rot(x, y, z : real) : real; // x1:=abs_rot(x0,y0,z0)
begin
abs_rot:=x*cos(teta)-y*sin(teta);
end;
{=====}
function ord_rot(x, y, z : real) : real; // y1:=ord_rot(x0,y0,z0)
begin
ord_rot:=x*sin(teta)+y*cos(teta);
end;
La cote reste inchangée //z1:=z0

```