

Exercice one : (Adchayan THAMBIAIAH)

We choose a custom format file called “.ksos”. The structure of the file will be like this :

Example.ksos

```
6 #number of points for the two curves

C1 #curve 1
1.0 -1.0 1.0 #each point of the curve on one line
-1.0 -1.0 1.0
...
end C1 #end of the curve 1

C2 #curve 2
1.0 -1.0 0.0
0.0 -1.0 1.0
...
End C2 #end of the curve 2
```

Pseudeo-code to implement:

Read_file_ksos(path):

Open the file in read mod

Read the first line => n_points

Init c1 = []

Init c2 = []

Mode = none

For each line in the file:

 If line is starting with “C1”:

 mode = c1

 else if line starting with “C2”:

 mode = c2

 else if starting with “end”:

 mode = aucun

 else if mode == c1:

 extract x,y,z and add (x,y,z) to c1

 else if mode == c2:

 extract x,y,z and add (x,y,z) to c2

```

    return c1,c2

triangulate_between_curve(c1,c2):
    init triangles = []
    for each i from 0 to len(c1)-2:
        p1 = c1[i]
        p2 = c1[i+1]
        q1 = courbe2[i]
        q2 = courbe2[i+1]

        triangle1 = (p1,p2,q1)
        triangle2 = (p2,q2,q1)
        add triangle1 to triangles
        add triangle2 to triangles
    return triangles

main():
    path = "/test/"
    c1,c2 = read_file_ksos(path)
    triangles = triangulate_between_curves(c1,c2)

```

Exercice two : (Mahamoud TOURE)

```

Generate_circle(R,n):
    Init cercle = []
    For each i from 0 to n-1:
        Angle = (2*pi*i)/n
        X = R*cos(angle)
        Y = R*sin(angle)
        Z = 0
        Add (x,y,z) to cercle
    Return circle

```

Rotate_curve(curve,angle_rad):

Init new_curve = []

For each (x,y,z) in curve:

$X_{new} = \cos(\text{angle_rad}) * x - \sin(\text{angle_rad}) * y$

$Y_{new} = \sin(\text{angle_rad}) * x + \cos(\text{angle_rad}) * y$

$Z_{new} = z$

Add (x_new,y_new,z_new) to new_curve

Return new_curve

Generate_copies_rotation(curve_profile,n):

Init curves_rot = []

For each I from 0 to n:

$\text{Angle} = (2 * \pi * i) / n$

Curve_rot = rotate_curve_aroud(curve_profile,angle)

Add curve_rot to curves_rot

Return curves_rot

Triangulate_revolution(courbes_rot):

Init triangles = []

For each i from 0 to len(curves_rot)-2:

$C1 = \text{curves_rot}[i]$

$C2 = \text{curves_rot}[i+1]$

For each j from 0 to len(c1)-2:

$P1 = c1[j]$

$P2 = c1[j+1]$

$Q1 = c2[j]$

$Q2 = c2[j+1]$

Triangle1 = (p1,p2,q1)

Triangle2 = (p2,q2,q1)

Add triangle1,triangle2 to triangles

Return triangles

Main():

Path = “/test/”

C1,Curve_profile = read_file_ksos(path)

N = 5 #number of divisions

Curves_rot = generate_copies_rotation(curve_profile,n)

Triangles = triangulate_revolution(curves_rot)