4.1 Experiment 1: Parametric Curves

This assignment illustrates Module 3, and it serves a purpose to teach you how to visualize curves defined by parametric functions. To work on this assignment, you have to watch the following TEL lectures:

Module 1: Lecture 2 (Part 3) - Introduction to Computer Graphics and Foundation Mathematics {Rene Descartes and coordinate systems}
Module 3: Lecture 1 (Part 2/3) - Geometric Shapes: 2D Curves {straight-lines}
Module 3: Lecture 1 (Part 3/3) - Geometric Shapes: 2D Curves {straight-lines}
Module 3: Lecture 2 (Part 1/3) - Geometric Shapes: 2D Curves {circle}
Module 3: Lecture 2 (Part 2/3) - Geometric Shapes: 2D Curves {circle and beyond}
Module 3: Lecture 2 (Part 3/3) - Geometric Shapes: 2D Curves {ellipse and summary}
Module 3: Lecture 3 - Geometric Shapes: 3D Curves

<u>Assignment instructions</u>:
Create folder **Lab1**. Download into it from the course-site the files **ParametricCurve.wrl** (Fig. 4) and **CoordinateAxes.wrl**. Use **ParametricCurve.wrl** as a template for the following exercises. For each of the curves, you have to select a minimally sufficient sampling resolution providing for smooth curve visualization so that any further reduction of it will visually reveal the polyline interpolation of the curve.

1. Define parametrically in 4 separate files using functions $x(u), y(u), u \in [0,1]$ and display:

    a. Straight line segment spanning from the point with coordinates (-***N***,-***M***) to the point with coordinates (***M***,***N***).

    b. A circular arc with radius ***N***, centered at point with coordinates (***N,M***) with the angles $[\frac{\pi}{\boldsymbol{N}}, 2\pi]$.

    c. Origin-centered 2D spiral curve which starts at the origin, makes ***N***+***M*** revolutions clockwise and reaches eventually the radius 2****M***.

    d. 3D cylindrical helix with radius ***N*** which is aligned with axis *Z*, makes ***M*** counterclockwise revolutions about axis *Z* while spanning from $z_1 = -\boldsymbol{N}$ to $z_1 = \boldsymbol{M}$.

(4 marks)

2. With reference to Table 1, convert the explicitly defined curve number ***M*** to parametric representations $x(u), y(u), u \in [0,1]$ and display it. Note that sketches of the curves in Table 1 are done not to the actual scale since the values of ***N*** and ***M*** are different in each variant.

(4 marks)

3. With reference to Figure 5, a curve is defined in polar coordinates by:

$$r = \boldsymbol{N} - (\boldsymbol{M} + 5)\cos\alpha \qquad \alpha \in [0,2\pi]$$

Define the curve parametrically as $x(u), y(u), u \in [0,1]$ and display it.

(4 marks)

4. Besides the above compulsory part, you are welcome to add any other shapes of parametric curves into folder Lab1/Extras. These extra exercises may increase your total mark.

```
#VRML V2.0 utf8
EXTERNPROTO FGeometry [
        exposedField SFString definition
        exposedField MFFloat parameters
        ……………………………………..
[ the rest of the EXTERNPROTO is skipped]

# External VRML object "Coordinate Axes" is included in the scene.
# The size of the axes can be changed by the scale transformation
Transform {
            scale 1.2 1.2 1.2 children [
                  Inline {url "CoordinateAxes.wrl"}  ]}

FShape {
# This definition is needed for drawing curves
polygonizer     "analytical_curve"

geometry FGeometry {
# The parametric formulae defining the curve. Change them to other formulae
# to see how geometry changes within the parameter domain
# and based on the sampling resolution defined below
definition "      x=1*(cos(2*pi*u))^3;
                  y=1*(sin(2*pi*u))^3;
                  z=0;"

# Domain for the parameter u.
# Explore how the curve changes when you change the domain values.
parameters [0 1]
# Sampling resolution along the curve. It defines how many times the parameter domain is
# sampled to calculate the curve function.
# Explore how the shape and the rendering speed change when you reduce or increase
# the resolution.
resolution [100]
 }
appearance FAppearance {
material FMaterial {
# Fixed red color is defined for the curve.
diffuseColor "r=1; g=0; b=0;"
  }  }
}
```

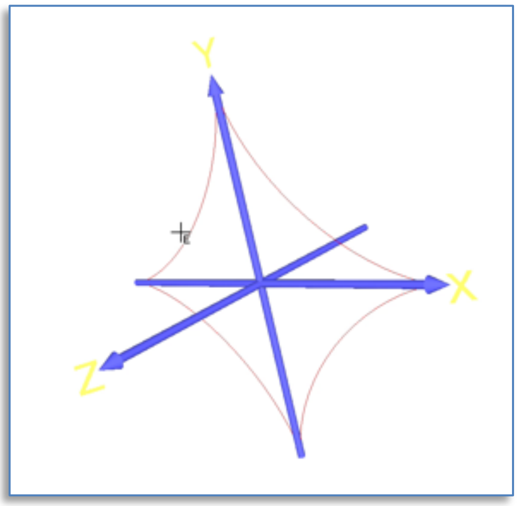
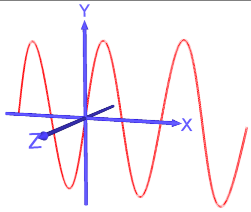
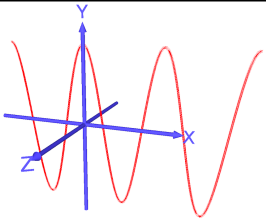
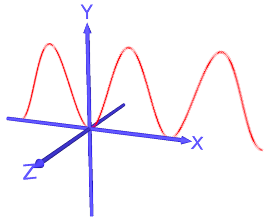
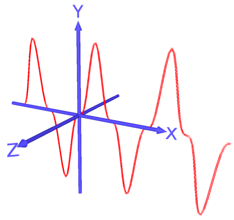
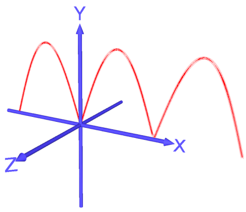
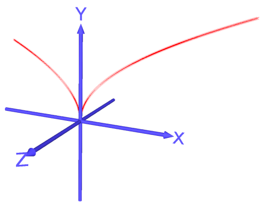
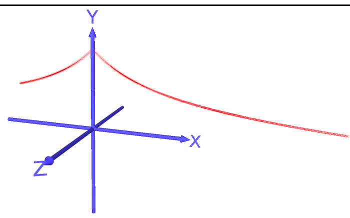
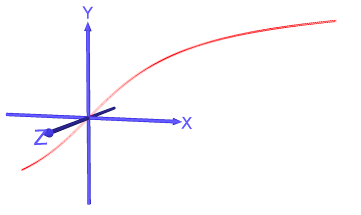
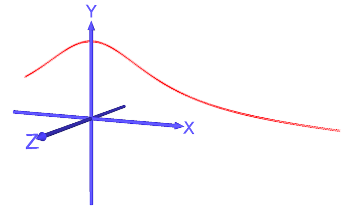
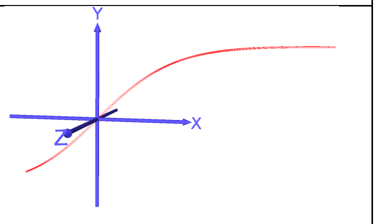


Figure 4. FVRML template of parametric curve. The code is in ParametricCurve.wrl.

Table 1. Curves defined explicitly.

M	Explicit formula	Sketch (not to the scale and with 3 full oscillations)
1	$y = \sin x$ The curve has to make ***N*** full oscillations within $x \in [-N, 2N]$	
2	$y = \cos x$ The curve has to make ***N*** full oscillations within $x \in [-N, N]$	
3	$y = (\sin x)^2$ The curve has to make ***N*** full oscillations within $x \in [-2N, N]$	
4	$y = (\sin x)^3$ The curve has to make ***N*** full oscillations within $x \in [-N, 3N]$	
5	$y = \lvert\sin x\rvert$ The curve has to make ***N*** full oscillations within $x \in [-1.5N, 2N]$	
6	$y = \sqrt{\lvert x\rvert} \quad x \in [-N, 1.5N]$	
7	$y = \dfrac{N}{\sqrt{\lvert x\rvert + 1}} \quad x \in [-1.5N, 2.5N]$	

8	$y = atan\, x \quad x \in [-0.7N, 2N]$	
9	$y = \dfrac{N}{x^2+1} \quad x \in [-N, 1.8N]$	
10	$y = \tanh x \quad x \in [-1.3N, 2N]$	

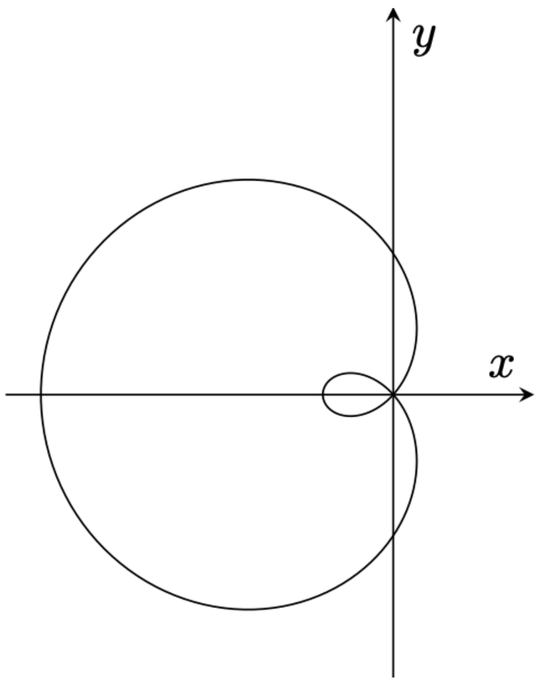


Figure 5. A polar curve called Limaçon $r = N - (M+5)\cos\alpha \quad \alpha \in [0, 2\pi]$. The actual shape of the curve is determined by the values of *N* and *M*.