CMPS 373 - Computer Graphics
Homework assignment number 2
To be submitted, through the blackboard system, before Sunday May 28, 2017

Bézier curves are a class of parametric curves defined using a set of control points, and a parameter t in [0,1]. The analytic parametric equation of a Bézier curve defined by a control polygon of n+1 points $P_i \;\; i = 0,\cdots,n$ is given through Bernstein polynomials $B_i^n(t)$ as follows: $C(t) = \sum_{i=0}^{n} B_i^n(t) P_i$ Where $B_i^n(t) = \frac{n!}{i!(n-i)!}(1-t)^{n-i} t^i$ is the i[th] Bernstein polynomial of degree n.
Given a value of parameter t, the Bézier curve can be evaluated either using the above parametric equation or using the Decasteljau subdivision algorithm.

The objective of this homework is to develop a tool for visualization and editing of Bézier curves of any degree n. A C++ class should be defined to encapsulate the data and all the functions necessary for easy manipulation of Bézier curves. A particular attention should be paid to the design of your application and to the graphic user interface that allows the activation of the desired functionality.

Recommended functionalities:

1. Appropriate data structures to store the curve properties (number of control points, set of control points, etc.)
2. A Menu to allow navigation through the application functionalities and select the function to launch.
3. Mouse control function: allows the definition of the control polygon, the selection and move of one control point into another position to enhance the curve design
4. A function BezPoint() to compute and return the point on the Bézier curve using the parametric equation.
5. A function CasteljauPoint() to compute and return a point on the Bézier curve using the Decasteljau algorithm.
6. A function CasteljauSubdivid() to compute and return the two sub-curves of a Bézier curve using the Decasteljau algorithm.
7. A function drawBez() to visualize a Bézier curve with (or without) its control polygon.
8. A function to edit the curve: use the mouse to select and move one control point into another position, then trace the resulting curve.

Remarks:

- Pay attention to the design of a clean, well-organized and efficient program.
- Added any function and/or data structure, not mentioned above but needed to enhance your application.
- Identical or similar applications will be ignored. Submission of source code uploaded from Internet is not allowed.

How to submit: Comment and organize your program in one single .cc file, then upload this file on the blackboard. email attached submissions will not be considered.

Deadline: The homework should be submitted before **Sunday May 28, 2017**. The blackboard system will not allow any submission after this date.