Министерство образования и науки Российской Федерации Федеральное агентство по образованию

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчет по лабораторной работе №2 дисциплины

«Компьютерная графика»

Выполнил студент группы ИВТ-22 /Крючков И. С/ Проверил /Коржавина А. С./

Киров 2021

**Цель работы:**

# Закрепить лекционный материал по изучению материала темы "Кривые Безье". Реализовать геометрический алгоритм вывода кривых Безье.

# Задание:

1. Написать на языке PASCAL программу, реализующую геометрический алгоритм построения кривой Безъе. Кривая должна строиться пошагово (с задержкой), отображая вспомогательные многоугольники, используемые для получения каждой точки.

2. Реализовать демонстрацию анимации движения объектов по сложным траекториям, составленным из кривых Безье.

**Теория:**

**Сплайн Безье**

Разработан математиком Пьером Безье. Кривые (сплайны) и поверхности Безье были использованы в 60-х годах компанией "Рено" для компьютерного проектирования формы кузовов автомобилей. В настоящее время они широко используются в компьютерной графике. Кривые Безье описываются в параметрической форме:

x=Px(t),

y=Py(t).

Значение t выступает как параметр, которому отвечают координаты отдельной точки линии. Параметрическая форма описания может быть более удобной для некоторых кривых, чем задание в виде функции у=f(х). Это потому, что функция f(х) может быть намного сложнее, чем Px(t) и Py(t), кроме того, f(х) может быть неоднозначной.

Многочлены Безье для Px(t) и Py(t) имеют такой вид:

где - сочетание m по i (известное также по биному Ньютона),

а хi и yi координаты точек-ориентиров Рi.

Значение m можно рассматривать и как степень полинома, и как значение, которое на единицу меньше количества точек-ориентиров.

**Геометрический алгоритм для кривой Безье**

1. Каждая сторона контура многоугольника, проходящего по точкам-ориентирам, делится пропорционально значению t.

2. Точки деления соединяются отрезками прямых и образуют новый многоугольник. Количество узлов нового контура на единицу меньше, чем количество узлов предыдущего контура.

3. Стороны нового контура снова делятся пропорционально значению t.

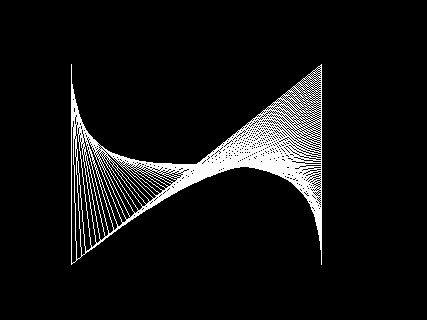
И так далее. Это продолжается до тех пор, пока не будет получена единственная точка деления. Эта точка и будет точкой кривой Безье.

**Схема алгоритма:**



Рисунок 1 – Схема алгоритма

**Экранная форма:**

****

**Листинг кода:**

program kg2;

uses graph, math, sysutils;

type

XY = record

x,y:integer;

end;

const

m = 4;

var

Gm, Gd, i, j, xn, yn:integer;

t,step:real;

R, P:array[1..m] of XY;

begin

Gd := Detect;

InitGraph(Gd, Gm, '');

P[1].x := 150;

P[1].y := 400;

P[2].x := 150;

P[2].y := 600;

P[3].x := 400;

P[3].y := 400;

P[4].x := 400;

P[4].y := 600;

xn := P[1].x;

yn := P[1].y;

t := 0;

step := 0.01;

repeat

R := P;

for j := m downto 2 do

begin

for i := 1 to j - 1 do

begin

R[i].x := R[i].x + round(t \* (R[i+1].x - R[i].x));

R[i].y := R[i].y + round(t \* (R[i+1].y - R[i].y));

end;

end;

for i := 1 to m - 1 do begin

line(R[i].x, R[i].y, R[i+1].x, R[i+1].y);

end;

sleep(10);

line(xn, yn, R[1].x, R[1].y);

t := t + step;

xn := R[1].x;

yn := R[1].y;

until t > 1;

readln()

end.

**Вывод:** в ходе выполнения лабораторной работы был закреплен лекционный материал по изучению материала темы "Кривые Безье". Был реализован геометрический алгоритм вывода кривых Безье с построением дополнительных линий.