**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CƠ SỞ**

**TÊN ĐỀ TÀI : Xây dựng và tính toán tập hợp điểm của đường cong tham số B-spline không đồng nhất**

Người hướng dẫn**: NGUYỄN TẤN KHÔI**

Sinh viên thực hiện**:**

Tên sinh viên**: Nguyễn Minh Quân**

Lớp**: 19TCLC\_Nhat1**

**Đà Nẵng, 07/2020**

MỤC LỤC

[MỤC LỤC i](#_Toc37682157)

[DANH MỤC HÌNH VẼ ii](#_Toc37682158)

[LỜI MỞ ĐẦU 1](#_Toc37682159)

[1. TỔNG QUAN ĐỀ TÀI 2](#_Toc37682160)

[2. CƠ SỞ LÝ THUYẾT 2](#_Toc37682161)

[2.1. Ý tưởng 2](#_Toc37682162)

[2.2. Cơ sở lý thuyết 2](#_Toc37682163)

[3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 2](#_Toc37682164)

[3.1. Phát biểu bài toán 2](#_Toc37682165)

[3.2. Cấu trúc dữ liệu 2](#_Toc37682166)

[3.3. Thuật toán 2](#_Toc37682167)

[4. CHƯƠNG TRÌNH VÀ KẾT QUẢ 2](#_Toc37682168)

[4.1. Tổ chức chương trình 2](#_Toc37682169)

[4.2. Ngôn ngữ cài đặt 2](#_Toc37682170)

[4.3. Kết quả 2](#_Toc37682171)

[4.3.1. Nhận xét đánh giá 3](#_Toc37682174)

[5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 3](#_Toc37682175)

[5.1. Kết luận 3](#_Toc37682176)

[5.2. Hướng phát triển 3](#_Toc37682177)

[TÀI LIỆU THAM KHẢO 4](#_Toc37682178)

[PHỤ LỤC 5](#_Toc37682179)

DANH MỤC HÌNH VẼ

* Hình 2.2.4.2: Hình ảnh đường cong B-spline không đồng nhất có bậc p = 2, có 4 đỉnh P0(10.0, 0.0), P1(20.0, 30.0), P2(50.0, 40.0), P3(80.0, 0.0); véctơ nút U = {0.0, 0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0, 1.0 } được xây dựng bằng <http://nurbscalculator.in> với điểm trên đường cong tham số với tham số u=0
* Hình 2.2.4.2 Hình ảnh đường cong B-spline không đồng nhất có bậc p = 2, có 4 đỉnh P0(10.0, 0.0), P1(20.0, 30.0), P2(50.0, 40.0), P3(80.0, 0.0); véctơ nút U = {0.0, 0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0, 1.0 } được xây dựng bằng <http://nurbscalculator.in> với điểm trên đường cong tham số với tham số u=0.25
* Hình 2.2.4.2: Hình ảnh đường cong B-spline không đồng nhất có bậc p = 2, có 4 đỉnh P0(10.0, 0.0), P1(20.0, 30.0), P2(50.0, 40.0), P3(80.0, 0.0); véctơ nút U = {0.0, 0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0, 1.0 } được xây dựng bằng <http://nurbscalculator.in> với điểm trên đường cong tham số với tham số u=1

LỜI MỞ ĐẦU

1. Mục đích thực hiện đề tài và mục tiêu đề tài

Công nghệ thông tin đang ngày càng phát triển và đang dần đi vào mọi mặt đời sống và xã hội. Cùng với đó là nhu cầu về đồ họa máy tính cũng tăng cao, đã tạo đà để phát triển công nghệ này phát triển nhanh chóng. Đồ họa đã được áp dụng rộng rãi khắp các lĩnh vực khác nhau từ công nghệ, y tế, kỹ thuật đến giải trí…

Đồ họa máy tính phát triển dựa trên các kết quả hình học họa hình, hình học vi phân cùng với nhiều kết quả toán học đặc biệt khác bao gồm cả đại số và giải tích. Hiện nay, với sự phát triển mạnh mẽ của phần cứng máy tính, đồ họa cx phát triển nhanh vượt bậc, tuy vậy nền tảng cơ bản của lĩnh vực này vẫn là cơ sở mô hình hóa hình học. Có nhiều bài toán đặt ra trong đồ họa máy tính. Một trong những bài toán cơ bản của nó là xử lý các đường cong và mặt cong.

B-Spline là một dạng đường cong và mặt cong trong mô hình hóa hình học đã được nhiều nhà khoa học trên thế giới nghiên cứu.

Đề tài này tìm hiểu về B-Spline, từ đó đưa ra những ứng dụng trong đồ họa máy tính.

1. Đối tượng và phạm vi nghiên cứu

* Đối tượng: cơ sở mô hình hóa hình học, B-Spline, Ứng dụng B-Spline trong đồ họa.
* Phạm vi: Đề tài tập trung nghiên cứu lý thuyết về B-Spline không đồng nhất và xây dựng tập hợp điểm của nó.

1. Nội dung nghiên cứu
   * Tìm hiểu tổng quan về mô hình hóa hình học.
   * Tìm hiểu lý thuyết về đường cong B-Spline.
   * Xây dựng thuật toán tính toán tập hợp điểm của đường cong B-Spline không đồng nhất
2. Phương pháp nghiên cứu

* Nghiên cứu những tài liệu tham khảo có liên quan.
* Xây dựng thuật toán bằng ngôn ngữ C++.

# TỔNG QUAN ĐỀ TÀI

Đường cong tham số Bézier được công bố vào năm 1962 bởi kỹ sư Pierre Bézier người Pháp, nhằm phục vụ cho thiết kế thân ôtô tại hãng Renault. Các cơ sở toán học cho các đường cong Bézier là các đa thức Bernstein được công bố từ năm 1912, nhưng không được sử dụng cho đến khi công bố đường cong Bézier.

Việc phát triển NURBS(*Non-uniform rational B-spline)* được bắt đầu từ những năm [1950](https://vi.wikipedia.org/wiki/1950) bởi những kĩ sư cần các biểu diễn toán học chính xác của các bề mặt tự do như vỏ của [tàu thủy](https://vi.wikipedia.org/wiki/T%C3%A0u_th%E1%BB%A7y), bề mặt bên ngoài của tàu không gian vũ trụ, và vỏ ôtô, những thứ mà có thể sản xuất lại một cách chính xác bất cứ khi nào bằng kĩ thuật. Trước đó, loại bề mặt như vậy chỉ được biểu diễn bởi một mô hình tạo bởi một người thiết kế mỹ thuật công nghiệp.

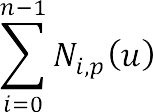
Những người tiên phong cho cuộc phát triển này có [Pierre Bézier](https://vi.wikipedia.org/wiki/Pierre_B%C3%A9zier), kỹ sư của [Renault](https://vi.wikipedia.org/wiki/Renault), và [Paul de Casteljau](https://vi.wikipedia.org/w/index.php?title=Paul_de_Casteljau&action=edit&redlink=1) làm việc cho [Citroën](https://vi.wikipedia.org/wiki/Citro%C3%ABn), cả hai đều ở [Pháp](https://vi.wikipedia.org/wiki/Ph%C3%A1p). Bézier đã làm việc gần như đồng thời với de Casteljau, không ai biết về công việc của người kia. Nhưng bởi vì Bézier đã công bố thành quả của ông ta, những người sử dụng công nghệ đồ họa máy tính ngày này công nhận đường Spline được biểu diễn với các điểm điều khiển giữ cữ cho đường cong là [đường cong Bézier](https://vi.wikipedia.org/wiki/%C4%90%C6%B0%E1%BB%9Dng_cong_B%C3%A9zier) splines, Trong khi tên tuổi của de Casteljau chỉ được biết và dùng cho thuật toán mà ông ấy phát triển để ước định bề mặt toán học điều khiển tham số. Vào những năm 1960 mọi việc đã trở nên rõ ràng rằng non-uniform rational B-splines là sự khái quát hóa của đường cong Bézier splines.

Với đường cong Bézier, ta có thể tạo ra các dạng đường cong khác nhau bằng cách hiệu chỉnh các đỉnh điều khiển cho tới khi có được dạng đường cong thoả mãn yêu cầu đặt ra ban đầu. Tuy nhiên, khi thay đổi bất kì một đỉnh điều khiển nào thì toàn bộ đường cong cũng sẽ bị thay đổi theo do tất cả các đa thức Bernstein đều khác 0 trên toàn đoạn [0, 1].

# CƠ SỞ LÝ THUYẾT

## Ý tưởng

Đường cong B-spline không đồng nhất là một đường cong tham số thường được sử dụng trong đồ họa máy tính, tính toán nội suy….. Một đường cong B- spline được xác định bằng tập hợp các điểm điều khiển *P0* đến *Pn* với *n* được gọi bậc của nó. Phương trình biểu diễn đường cong B-spline không đồng nhất bậc *n*:



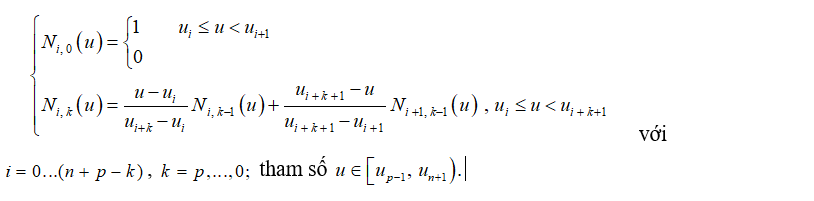
Tập vectơ nút *U = { u0 , … un + p } ui  ui+1*

* *p* là bậc (*degree*) của đường congvới *p,n∈Z*
* *Pi ∈R (i=0...n)*là tập các đỉnh kiểm điều khiển của đường cong;
* Vector nút *U* có *(n + p + 2)* phần tử là một dãy các giá trị số không giảm.

* ∀ ui ≤ui+1,ui∈R.*

## Cơ sở lý thuyết

### Công thức tổng quát

Các hàm cơ sở B-spline được xác định trong các khoảng giá trị của vector nút U. Hàm cơ sở thứ *i* bậc *p* ký hiệu *Ni,p* được biễu diễn như sau :

Để tránh trường hợp mẫu số bằng 0 khi , , ký hiệu . Mỗi hàm được xác định trên *k* khoảng cách của vector nút.

Đường cong B-spline được xác định trong đoạn tham số  và bao gồm nhiều đoạn cong, các đoạn cong này gặp nhau tại các điểm được xác định tương ứng với các giá trị của vector nút: **

### Vector nút không đồng nhất

Các nút trong vector nút có khoảng cách tuỳ ý, không giảm, có thể lặp giá trị lại nhiều lần

**

***vector nút khép***

Dạng vector nút này có các giá trị lặp lại tại *(p + 1)* phần tử đầu và   
*(p + 1)* phần tử cuối.



VD: *n =* 3*, p =* 2*,*  *U* = { *0, 0, 0, 0.7, 2, 2, 2* }

*n =* 4*, p =* 3*,*  *U* = { *0, 0, 0, 0, 0.5, 1, 1, 1, 1* }

*n =* 5*, p =* 4*,*  *U* = { *0, 0, 0, 0, 0, 1, 2, 2, 2, 2, 2* }

### Hàm cơ sở:

Các hàm cơ sở B-spline được xác định trong các khoảng giá trị của vector nút *U*. Hàm cơ sở thứ *i* bậc *p* ký hiệuđược định nghĩa dạng biểu thức đệ quy (theo de Boor, Cox, Mansfield).

Hàm cơ sở B-spline bậc *p*,chỉ phụ thuộc vào bậc *p* và vector nút *U*, được biểu diễn bằng cách kết hợp tuyến tính các hàm cơ sở B-spline có bậc thấp hơn.

Đặt , hàm cơ sở được biểu diễn dạng rút gọn:



Trong hệ toạ độ Cartesian, một điểm trên đường cong B-spline tương ứng với tham số *u* có toạ độ như sau:



### Tính triển khai ví dụ

Tính tọa độ điểm C(u) của đường cong B-spline không đồng nhất bậc *p*=2, có 4 đỉnh *P0(10.0, 0.0), P1(20.0, 30.0), P2(50.0, 40.0), P3(80.0, 0.0),* vector nút *U={0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0}*

Xác định các điểm của đường cong tham số ứng với *u=0;u=0,25;u=1.*

**Ta có công thức :**

với ,  tham số 

**Với u = 0**

*N0,0*(0)*=*1[*u*0*,u*1)=[0.0,0.0)

*N1,0*(0)*=*1[*u*1*,u*2)=[0.0,0.0)

*N2,0*(0)*=*1[*u*2*,u*3)=[0.0,0.5)

*N3,0*(0)*=*0[*u*3*,u*4)=[0.5,1.0)

*N4,0*(0)*=*0[*u*4*,u*5)=[1.0,1.0)

*N5,0*(0)*=*0[*u*5*,u*6)=[1.0,1.0)

Bậc k = 1:

*Ni,1*(0)*=Ni,0(u)+ Ni+1,0(u), ui≤u<ui+2*

*N0,1*(0)*=N0,0*(0)*+ N1,0*(0)

=×1+×1

=0

*N1,1*(0)*=N1,0*(0)*+ N2,0*(0)

=×1+×1

=1

*N2,1*(0)*=N2,0*(0)*+ N3,0*(0)

=×1+×0

=0

*N3,1*(0)*=N3,0*(0)*+ N4,0*(0)

=×0+×0

=0

*N4,1*(0)*=N4,0*(0)*+ N5,0*(0)

=×0+×0

=0

Bậc k = 2:

*Ni,2=Ni,1(u)+ Ni+1,1(u), ui≤u<ui+3*

*N0,2*(0)*=N0,1*(0)*+ N1,1*(0)

=×0+×1

=1

*N1,2*(0)*=N1,1*(0)*+ N2,1*(0)

=×1+×0

=0

*N2,2*(0)*=N2,1*(0)*+ N3,1*(0)

=×0+×0

=0

*N3,2*(0)*=N3,1*(0)*+ N4,1*(0)

=×0+×0 =0

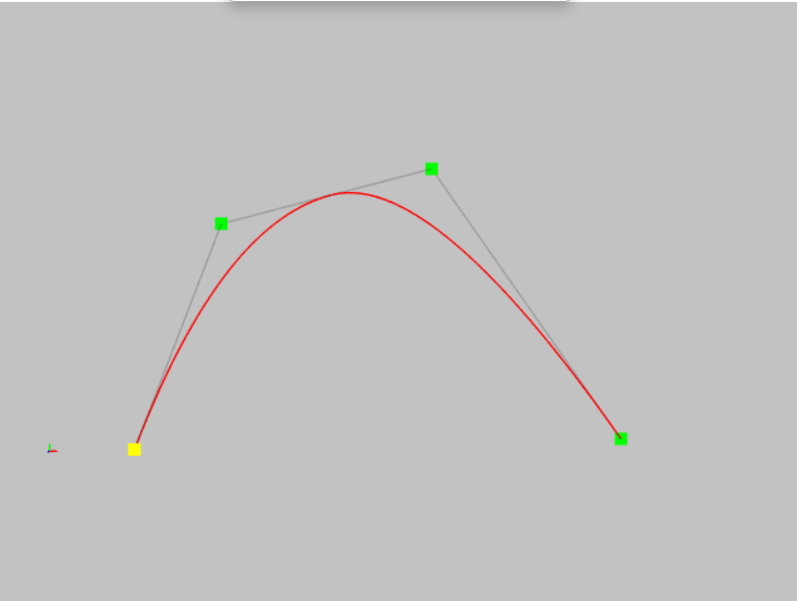
**Thay vào công thức:**





*C*(*u*)=1.0*P0*

Vậy kết quả điểm của đường cong tham số ứng với *u=0* là *C(0)=(10,0,0).*



*Hình 2.2.4.1: Điểm trên đường cong tham số với tham số u=0.*

**Với u = 0,25:**

*N0,0*(0.25)*=*0[*u*0*,u*1)=[0.0,0.0)

*N1,0*(0.25)*=*0[*u*1*,u*2)=[0.0,0.0)

*N2,0*(0.25)*=*1[*u*2*,u*3)=[0.0,0.5)

*N3,0*(0.25)*=*0[*u*3*,u*4)=[0.5,1.0)

*N4,0*(0.25)*=*0[*u*4*,u*5)=[1.0,1.0)

*N5,0*(0.25)*=*0[*u*5*,u*6)=[1.0,1.0)

Bậc k = 1:

*Ni,1*(0.25)*=Ni,0(u)+ Ni+1,0(u), ui≤u<ui+2*

*N0,1*(0.25)*=N0,0*(0.25)*+ N1,0*(0.25)

=×0+×0

=0

*N1,1*(0.25)*=N1,0*(0.25)*+ N2,0*(0.25)

=×0+×1

=0.5

*N2,1*(0.25)*=N2,0*(0.25)*+ N3,0*(0.25)

=×1+×0

=0.5

*N3,1*(0.25)*=N3,0*(0.25)*+ N4,0*(0.25)

=×0+×0

=0

*N4,1*(0.25)*=N4,0*(0.25)*+ N5,0*(0.25)

=×0+×0

=0

Bậc k = 2:

*Ni,2=Ni,1(u)+ Ni+1,1(u), ui≤u<ui+3*

*N0,2*(0.25)*=N0,1*(0.25)*+ N1,1*(0.25)

=×0+×0.5

=0.25

*N1,2*(0.25)*=N1,1*(0.25)*+ N2,1*(0.25)

=×0.25+×0.5

=0.625

*N2,2*(0.25)*=N2,1*(0.25)*+ N3,1*(0.25)

=×0.5+×0

=0.125

*N3,2*(0.25)*=N3,1*(0.25)*+ N4,1*(0.25)

=×0+×0 =0

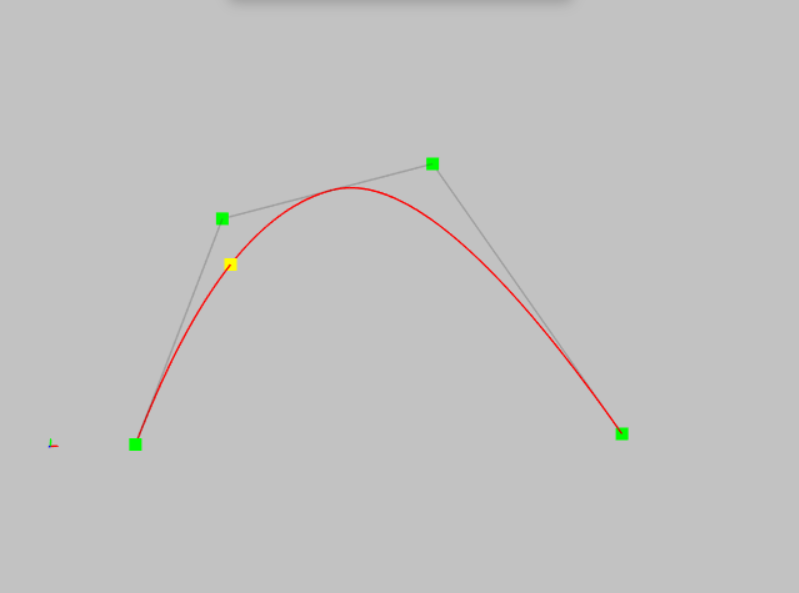
**Thay vào công thức:**





*C*(*u*)=0.25*P0*+0.625*P1*+0.125*P2*

Vậy kết quả điểm của đường cong tham số ứng với *u=0* là *C(0.25)=(21.75,23.75,0).*



*Hình 2.2.4.2: Điểm trên đường cong tham số với tham số u=0.25*

**Với u = 1**

*N0,0*(1)*=*0[*u*0*,u*1)=[0.0,0.0)

*N1,0*(1)*=*0[*u*1*,u*2)=[0.0,0.0)

*N2,0*(1)*=*0[*u*2*,u*3)=[0.0,0.5)

*N3,0*(1)*=*1[*u*3*,u*4)=[0.5,1.0)

*N4,0*(1)*=*1[*u*4*,u*5)=[1.0,1.0)

*N5,0*(1)*=*1[*u*5*,u*6)=[1.0,1.0)

Bậc k = 1:

*Ni,1*(1)*=Ni,0(u)+ Ni+1,0(u), ui≤u<ui+2*

*N0,1*(1)*=N0,0*(1)*+ N1,0*(1)

=×0+×0

=0

*N1,1*(1)*=N1,0*(1)*+ N2,0*(1)

=×0+×0

=0

*N2,1*(1)*=N2,0*(1)*+ N3,0*(1)

=×0+×0

=0

*N3,1*(1)*=N3,0*(1)*+ N4,0*(1)

=×1+×1

=1

*N4,1*(1)*=N4,0*(1)*+ N5,0*(1)

=×1+×1

=0

Bậc k = 2:

*Ni,2=Ni,1(u)+ Ni+1,1(u), ui≤u<ui+3*

*N0,2*(1)*=N0,1*(1)*+ N1,1*(1)

=×0+×0

=0

*N1,2*(1)*=N1,1*(1)*+ N2,1*(1)

=×0+×0

=0

*N2,2*(1)*=N2,1*(1)*+ N3,1*(1)

=×0+×0

=0

*N3,2*(1)*=N3,1*(1)*+ N4,1*(1)

=×1+×0

=1

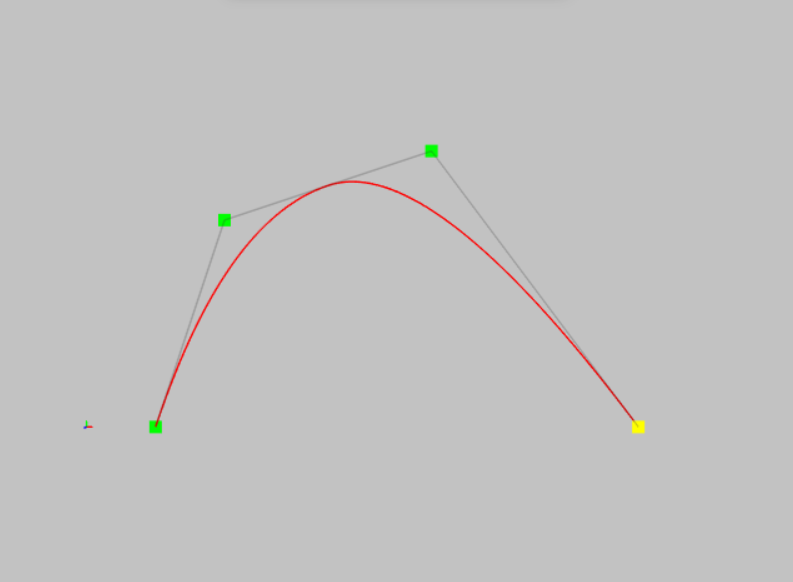
**Thay vào công thức:**





*C*(*u*)=1.0*P3*

Vậy kết quả điểm của đường cong tham số ứng với *u=1* là *C(1)=(80,0,0).*



*Hình 2.2.4.3: Điểm trên đường cong tham số với tham số u=1*

# TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

## Phát biểu bài toán

Tính tọa độ điểm C(u) (với giá trị tham số *u* tự chọn trong khoảng [0, 1]) của đường cong B-spline không đồng nhất bậc *p*, có n đỉnh *P0(x0, y0, z0), P1(x1, y1, z1),* …, *Pn(xn, yn, zn);* véctơ nút *U = {u1, u2, …, un+p+2 }*.

Input:

-Các giá trị *n, p, u*.

-Mỗi đỉnh thứ i chứa 3 giá trị *xi, yi, zi.*

-Giá trị *ui*.

Output:

Tọa độ các điểm *C(u) = (x, y, z)* tương ứng với các tham số *0≤u≤1* của đường cong B-spline.

## Cấu trúc dữ liệu

Sử dụng cấu trúc dữ liệu mảng động để lưu trữ và tính toán xử lý.

Mảng động là mảng có thể thay đổi kích thước của mảng bất cứ khi nào thêm hoặc xóa phần tử trong mảng.

Cấp phát động một mảng cho phép chúng ta xác định kích thước mảng trong thời gian chạy chương trình.

Để cấp phát và thu hồi mảng động, C++ cung cấp toán tử new[] và delete[]:

## Thuật toán

*Thuật toán:* Để xác định một điểm trên đường cong B-spline ứng với tham số *u*, ta triển khai tính toán đệ qui các đa thức *Ni,p(u)* và đa thức *C(u).*

## Tổ chức chương trình

Tính hàm cơ bản với BasisFuns

Tính khoảng điểm thỏa mãn

Tính điểm *C(u)*

## Ngôn ngữ cài đặt

Ngôn ngữ được sử dụng là C++

## Kết quả

Trả về giá trị *C(u)(x,y,z)* thỏa mãn với tham số u cho trước.

### Nhận xét đánh giá

### Kết quả trả về là một điểm trên đường cong B-spline không đồng nhất, dựa trên kết quả ta có thể xây dựng nên chương trình vẽ đường cong hoàn chỉnh bằng những ngôn ngữ bậc cao.

### Thuật toán BasicFunction

Bằng cách sử dụng hàm *FindSpan* để tìm thứ tự i theo tham số u và sử dụng cấu trúc mảng động để lưu trữ những hàm số cơ sở sẽ tiết kiệm dung lượng và thời gian xử lý.

Sử dụng vòng lặp để tính toán từng bậc k của các hàm số thứ i theo u ở vector nút U đã cho trước bằng cách áp dụng công thức:

*Ni,1*(0)*=Ni,k-1(u)+ Ni+1,k-1(u), ui≤u<ui+k+1*

Hàm đã được tính toán sẽ lưu trữ trong mảng động N để sử dụng trong thuật toán tính điểm.

### Thuật toán tính điểm CurvePoint3D

Sử dụng kết quả đã được tính toán nhờ thuật toán *BasicFunction*, áp dụng công thức:



Tính toán điểm *C(u)* và lưu kết quả vào mảng *C*.

### Kết quả thuật toán:

Một dòng kết quả là *C(u)=(x,y,z)*.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

B-spline được phát triển và sử dụng bởi nhiều ngành nghề trong suốt thời gian qua, việc xây dựng tập hợp điểm trên đường cong cho trước là bước cơ bản để từ đó hoàn thành những bước tiến quan trọng hơn.

## Hướng phát triển

Từ mô hình đường cong B-Spline, ta có thể xây dựng nên mặt cong B-Spline tương ứng, mở ra nhiều cơ hội cho việc phát triển các ngành đồ họa và đặc biệt là phát triển công nghệ mới.

Đường cong B-Spline đã đang và sẽ được ứng dụng rộng rãi hơn trong nhiều lĩnh vực, vì vậy việc áp dụng công nghệ thông tin để hỗ trợ tính toán và xây dựng mô hình đường cong B-Spline và mặt cong B-Spline là hoàn toàn thiết thực.

**TÀI LIỆU THAM KHẢO**

1. Tài liệu/slide về đường cong tham số do GV cung cấp.
2. Advanced Graphics and HCI - B-Splines, Dr Neil Dodgson, University of Cambridge Computer Laboratory, https://[www.cl.cam.ac.uk/teaching/1999/AGraphHCI/SMAG/node4.html,](http://www.cl.cam.ac.uk/teaching/1999/AGraphHCI/SMAG/node4.html) 23/10/2019.
3. Uniform Cubic B-Spline Curves, Howard J. Hamilton, University of Regina,<http://www2.cs.uregina.ca/~anima/408/Notes/Interpolation/UniformBSpline.htm>, 23/10/2019.
4. [http://nurbscalculator.in/,](http://nurbscalculator.in/) 21/11/2019.
5. https://vi.wikipedia.org/wiki/NURBS.

PHỤ LỤC

Thuật toán tìm hàm cơ bản (BASIC FUNCTION) trong *i* với tham số *u,* bậc *p*, vector nút *U*[]*,* kết quả trả về giá trị *N*[*p+*1]:

int FindSpan(int n, int p, double u, double \*U) {

int low, mid, high;

if(u >= U[n]) return n;

if(u <= U[p])return p;

low = 0; /\* low = p; \*/

high = n+1; mid = (low+high)/2;

while(u < U[mid] || u >= U[mid+1]){

if(u < U[mid]) high = mid;

else low = mid;

mid = (low + high)/2;

}

return(mid);

}

void BasisFuns(int i, double u, int p, double \*U, double \*N){

double \*left = NULL, \*right = NULL, saved, temp;

int j, r;

if(!(left = (double\*)calloc(p+1, sizeof(double)))) return;

if(!(right=(double\*)calloc(p+1, sizeof(double)))) { free(left); return; }

N[0] = 1.0;

for(j = 1; j <= p; j++) {

left[j] = u - U[i+1-j];

right[j] = U[i+j] - u;

saved = 0.0;

for(r = 0; r < j; r++){

temp = N[r] / (right[r+1] + left[j-r]);

N[r] = saved + right[r+1] \* temp;

saved = left[j-r] \* temp;

}

N[j] = saved;

}

free(left);

free(right);

}

**Thuật toán tính điểm CurvePoint3D trên đường cong B-spline bậc *p* tương ứng với tham số *u* trong vector nút *U:***

int CurvePoint3D(int n, int p, double \*U, double \*P, double u, double \*C){

int span, j, k;

double \*N = NULL;

if(!(N = (double\*)calloc(p+1, sizeof(double)))) return 0;

span = FindSpan(n, p, u, U);

BasisFuns(span, u, p, U, N);

C[0] = 0.0; C[1] = 0.0; C[2] = 0.0;

for(j = 0; j <= p; j++){

k = (span-p+j)\*3;

C[0] = C[0] + N[j]\*P[k];

C[1] = C[1] + N[j]\*P[k+1];

C[2] = C[2] + N[j]\*P[k+2];

}

free(N);

return 1;

}

**Chương trình chính:**

int main(){

std::ifstream fileInput("C:/Users/Admin/Desktop/example.txt");

//Duong dan den file vi du duoc neu trong bao cao

int n,p; double u;

double \*U=new double[n+p+2];

double \*P=new double[(n+1)\*3];

double C[3];

if (fileInput.fail())

{

std::cout << "Failed to open this file!" << std::endl;

return 1;

}

cout<<"Nhap tham so u: ";

cin>>u;

while (!fileInput.eof())

{

fileInput >> n >> p;

P=(double\*)calloc(n+1, sizeof(double));

U=(double\*)calloc(n+p+2, sizeof(double));

for(int i=1;i<=n+1;i++){

int k=i\*3;

fileInput>>P[k]>>P[k+1]>>P[k+2];

}

for(int i=1;i<=(n+p+2);i++){

fileInput>>U[i];

}

}

fileInput.close();

CurvePoint3D(n, p, U, P, u, C);

cout<<"C(u)="<<"("<<C[0]<<","<<C[1]<<","<<C[2]<<")";

return 0;

}