**LỜI NÓI ĐẦU**

Trong xu thế tin học hóa toàn bộ cuộc sống đang diễn ra tấp nập hiện nay, lĩnh vực đồ họa đóng góp một vai trò cực kỳ to lớn. Ở đâu ta cũng có thể tìm thấy những minh họa cho việc áp dụng đồ hạo máy tính vào cuộc sống: xử lý ảnh, phim hoạt hình, nghiên cứu, xây dựng và tập hợp các công cụ (mô hình lý thuyết và phần mềm) khác nhau để *kiến tạo, xây dựng, lưu trữ và xử lý* các *mô hình* và *hình ảnh* của các đối tượng, sự vật, hiện tượng trong cuộc sống, sản xuất, nghiên cứu.

Đồ Họa Máy Tính góp phần quan trọng làm cho giao tiếp giữa con người và máy tính trở nên thân thiện hơn. Trong lĩnh vực kỹ thuật máy tính, Đồ Họa Máy Tính càng ngày càng phát triển mạng mẽ. Từ đồ họa trên máy tính chúng ta có nhiều lĩnh vực có ứng dụng rất quan trọng của Đồ Họa Máy Tính trong thực tế như: tạo mô hình, hoạt cảnh (game, giải trí,…), hỗ trợ thiết kế đồ họa, mô phỏng hình ảnh, chuẩn đoán hình ảnh (trong Y tế), huấn luyện đào tạo ảnh (quân sự, hàng không,…)

Trong Đồ Họa Máy Tính có nhiều thuật toán khác nhau. Xong mỗi thuật toán lại tỏ ra có những ưu việt và hạn chế riêng đối với từng bài toán cụ thể. Và để phục vụ cho việc tìm hiểu thêm về các thuật toán trong Đồ Họa Máy Tính, Nhóm chúng em với sự hướng dẫn của Giảng viên – – Khoa CNTT – Trường ĐH Điện Lực đã thực hiện đề tài: *“Viết chương trình cài đặt thuật toán Bresham. Ứng dụng thuật toán Bresham để vẽ ngôi nhà”.*

Chúng em xin gửi lời cảm ơn chân thành tới thầy giáo: *“***……….***”* đã tận tâm chỉ bảo và giúp đỡ chúng em hoàn thành đề tài này. Trong quá trình làm đề tài sẽ không tránh khỏi những sai sót, khuyết điểm. Vì vậy, nhóm thực hiện chúng em hy vọng nhận được sự đánh giá và đóng góp nhiệt tình từ phía thầy cô và các bạn để bài của nhóm chúng em được hoàn thiện hơn

***Chúng em xin chân thành cảm ơn!***

Trường ĐH Điện Lực, tháng 04 năm 2020.

**PHẦN 1: GIỚI THIỆU CHUNG VỀ ĐỀ TÀI**

## Giới Thiệu Đề Tài.

Cùng với sự phát triển của nền kinh tế hiện nay là sự phát triển mạnh mẽ của ngành công nghệ thông tin. Các ứng dụng của công nghệ thông tin ngày càng nhiều và có mặt trong hầu hết các lĩnh vực của cuộc sống.

Với đề tài: *“Viết chương trình cài đặt thuật toán Bresenham. Ứng dụng thuật toán Bresenham để vẽ ngôi nhà”*, nhóm Vinh + Hiếu+ Lâm – lớp D13CNPM2 sẽ trình bày về thuật toán Bresham. Ý nghĩa của thuật toán Bresham trong công nghệ thông tin là không hề nhỏ, đặc biệt là đối với màn hình máy tính thì nó có ý nghĩa vô cùng quan trọng.

Qua đề tài này nhóm, mong sẽ giúp ích được cho mọi người hiểu thêm phần nào về thuật toán Bresham.

1. **Giới Thiệu Ngôn Ngữ Lập Trình Được Sử Dụng Trong Đề Tài**

Stroustrup đã bắt đầu làm việc với khái niệm lớp trong [1979](http://vi.wikipedia.org/wiki/1979). Ý tưởng tạo ra một ngôn ngữ mới bắt nguồn từ kinh nghiệm lập trình khi mà ông viết luận án tiến sĩ. Stroustrup nhận ra rằng [Simula](http://vi.wikipedia.org/w/index.php?title=Simula&action=edit&redlink=1)có nhiều tính năng hữu dụng cho việc phát triển một phần mềm lớn nhưng nó đã quá chậm trong ứng dụng thực tế, trong khi đó, [BCPL](http://vi.wikipedia.org/w/index.php?title=BCPL&action=edit&redlink=1) lại nhanh nhưng ở cấp quá thấp và không tiện cho việc phát triển phần mềm lớn. Đến khi làm việc ở [Bell Labs](http://vi.wikipedia.org/w/index.php?title=Bell_Labs&action=edit&redlink=1), thì ông gặp phải vấn đề trong việc phân tích [nhân Unix](http://vi.wikipedia.org/w/index.php?title=Nh%C3%A2n_Unix&action=edit&redlink=1) với việc [tính toán phân tán](http://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_to%C3%A1n_ph%C3%A2n_t%C3%A1n). Dùng lại kinh nghiệm lúc làm luận án tiến sĩ, Stroustrup cài thêm các tính năng giống như Simula vào trong C để nâng cao. C được chọn là vì đó là ngôn ngữ tổng quát, nhanh và năng động. Lần đầu tiên, các chức năng như là lớp, lớp dẫn xuất, kiểm tra kiểu mạnh, nội tuyến (inline), và đối số mặc định đã được thêm vào trong C. Lần xuất bản đầu tiên vào thị trường xảy ra trong tháng 11/[1985](http://vi.wikipedia.org/wiki/1985).

Năm [1983](http://vi.wikipedia.org/wiki/1983), thì tên *C với các lớp* được đổi thành C++. các chức năng mới được thêm vào bao gồm hàm ảo, quá tải hàm và toán tử, tham chiếu, hằng, khả năng kiểm soát bộ nhớ của lưu trữ tự do, nâng cao việc kiểm soát kiểu, và lệnh chú giải kiểu (//).

Năm [1985](http://vi.wikipedia.org/wiki/1985), tác phẩm *The C++ Programming Language* được xuất bản lần đầu tiên, cung cấp một tài liệu tham khảo quan trọng cho ngôn ngữ nhưng đó chưa là một tiêu chuẩn chính thức.

Năm [1989](http://vi.wikipedia.org/wiki/1989) phiên bản C++ 2.0 phát hành. Các tính năng mới bao gồm đa kế thừa, lớp trừu tượng, hàm tĩnh, hàm thành viên hằng, và thành viên bảo tồn.

Năm [1990](http://vi.wikipedia.org/wiki/1990), cuốn *The Annotated C++ Reference Manual* được xuất bản cung cấp nền tảng cho tiêu chuẩn tương lai.

Phiên bản xuất bản sau đó có thêm các chức năng tiêu bản, ngoại lệ, không gian tên, chuyển kiểu cho toán tử new, và kiểu Boolean.

Khi C++ hình thành, thì thư viện chuẩn hoàn thiện với nó. Thư viện C++ đầu tiên thêm vào là iostream.h cung cấp cơ sở để thay thế các hàm C truyền thống như là printf và scanf. Sàu này, trong những thư viện chuẩn quan trọng nhất được thêm vào là [Thư viện Tiêu bản Chuẩn](http://vi.wikipedia.org/w/index.php?title=Th%C6%B0_vi%E1%BB%87n_Ti%C3%AAu_b%E1%BA%A3n_Chu%E1%BA%A9n&action=edit&redlink=1).

Sau nhiều năm làm việc, có sự cộng tác giữa [ANSI](http://vi.wikipedia.org/wiki/ANSI) và hội đồng tiêu chuẩn hoá C++ của [ISO](http://vi.wikipedia.org/wiki/ISO) để soạn thảo tiêu chuẩn [*ISO/IEC 14882*](http://vi.wikipedia.org/w/index.php?title=ISO/IEC_14882&action=edit&redlink=1)*:1998*. Phiên bản tiêu chuẩn này được phát hành năm [1989](http://vi.wikipedia.org/wiki/1989), hội đồng tiếp tục xử lí các báo cáo trục trặc, và ấn hành một phiên bản sửa sai của chuẩn C++ trong năm 2003.

Không ai là chủ nhân của ngôn ngữ C++, nó hoàn toàn miễn phí khi dùng. Mặc dù vậy, các văn bản tiêu chuẩn thì không miễn phí.

## III. Giới Thiệu Thư Viện Graphics.h Trong Dev-C++.

Tạo một chương trình đồ hoạ 2D trong DOS sữ dụng turbo C khá dễ dàng và ai cũng có thể làm được.Thư viện đựoc #include có tên là graphis.h. Nhưng các thư viện C/C++ của borland lại có 1 điều đặc biệt là không sữ dụng được đối với 1 số trình compiler khác, tuy nhiên so với các IDE hiện đại thì borland lại kém về mặt quản lý và tính thuận tiện.

Chính vì lẽ đó mà bản hack thư viện graphics cũng như các thư viện khác ra đời, và hôm nay thư viện mình muốn giới thiệu graphics.h của Micheal.

Micheal đã thay đỗi BGI library (thư viện BGI) thành thư viện có tên WinBGIm để có thể sữ dụng tốt trên windows. Và bây giờ bạn đã có thể sữ dụng tốt các hàm đặc biệt của borland bằng dev-c++

**IV.** **Cách sử dụng thư viện đồ họa trong Dev-C++.**

Đầu tiên bạn tải file đính kèm về máy mình, giải nén ra sẽ được 2 file là graphics.h và libbgi.a  
+ File graphics.h để ở thư mục C:\Dev-Cpp\include  
+ File libbgi.a để ở thư mục C:\Dev-Cpp\lib  
(Nếu bạn cài mặc đinh Dev C++ ở ổ C)  
  
Tiếp theo bạn khởi động Dev C++ lên, vào File-->New--> Project...-->Empty Project (Nhớ chọn C++ Project) -->OK  
Đặt tên cho Project của mình!  
  
Nhấn chuột phải lên cái project của bạn -->New File  
Nhấn Alt + P --->Chọn thẻ Parameters--> Copy và paste vào ô Linker những dìng dưới đây:  
-lbgi  
-lgdi32  
-lcomdlg32  
-luuid  
-loleaut32  
-lole32  
(Bao gồm cả dấu "-")  
-->OK

**Phần 2: Ý tưởng xây dựng chương trình và cài đặt thuật toán**

**I.Thuật toán Bresenham**

* 1. **thuật toán vẽ đoạn thẳng**

Thuật toán Bresenham vẽ đoạn thẳng trường cho trường hợp hệ số góc 0<m<1 có thể mô tả

tóm tắt như sau:

• **Bước 1**: Nhập các điểm đầu mút. Điểm đầu mút bên trái chứa tọa độ (x1,y1), điểm đầu

mút bên phải chứa tọa độ (x2,y2).

• **Bước 2**:

- Tính Δx = |x2 - x1| , Δy = |y2 - y1| và p = 2Δy - Δx.

- Chọn điểm xuất phát (x,y) = (x1,y1).

- Vẽ điểm (x,y).

• **Bước 3: Nếu x<x2 thì x = x+1.**

* 1. - Nếu p < 0: p = p + 2Δy

Ngược lại: p = p + 2( Δy - Δx) và **y = y+1**.

- Vẽ điểm (x,y) mới.

• **Bước 4: Lặp lại bước 3 cho đến khi x = x2.**

Sau đây là hàm cài đặt thuật toán:

void Line(int x1,int y1,int x2,int y2)

// 0<m<1

{

int x\_max,x,y;

int dx=abs(x1-x2);

int dy=abs(y1-y2);

int c1=2\*dy;

int c2=2\*(dy-dx);

int p=2\*dy-dx;

if (x1>x2)

{

x=x2; y=y2; x\_max=x1;

}

else

{

x=x1; y=y1; x\_max=x2;

}

putpixel(x,y,color);

while (x < x\_max)

{

x=x+1;

if (p<0) p=p+c1;

else

{

y=y+1;

p=p+c2;

}

putpixel(x,y,5);

}

}

**1.2 Thuật toán vẽ đường tròn**

Giả sử đã vẽ được điểm (xi,yi), điểm kế tiếp cần vẽ là (xi+1,yi) hoặc (xi+1,yi-1) (Hình 1.5).

Từ phương trình: x2 + y2 = R2, giá trị y thực ứng với xi +1 sẽ là:

y2 = R2 - (xi +1)2

Đặt: d1 = yi

2 - y2 = yi

2 - R2 + (xi + 1)2

d2 = y2 - (yi - 1)2 = R2 - (xi + 1)2 - (yi - 1)2



Suy ra:

pi = d1 - d2 = 2.(xi + 1)2 + yi

2 + (yi - 1)2 - 2R2 (2)

⇒ pi+1 = 2.(xi+1 + 1)2 + y2

i+1 + (yi+1 - 1)2 - 2R2 (3)

Lấy (3) - (2), ta có:

pi+1 - pi = 4xi + 6 + 2.(y2

i+1 - yi

2) - 2.(yi+1 - yi)

⇒ **pi+1 = pi + 4xi + 6 + 2.(y2**

**i+1 - yi**

**2) - 2.(yi+1 - yi)** (4)

***\* Nhận xét***:

• Nếu pi < 0: chọn **yi+1 = yi** (4) ⇒ **pi+1 = pi + 4xi + 6**

• Nếu pi ≥ 0: chọn **yi+1 = yi - 1** (4) ⇒ **pi+1 = pi + 4.(xi - yi) + 10**

• Ta chọn điểm đầu tiên cần vẽ là (0,R), theo (2): **p1 = 3 - 2R.**

Thuật toán Bresenham để vẽ đường tròn được phát thảo như sau:

• **Bước 1**:

- Chọn điểm đầu cần vẽ (x,y) = (0,R).

- Tính p đầu tiên: p = 3 - 2R.

- Vẽ 8 điểm ứng với (x,y).

• **Bước 2**:

- Tăng x lên 1 pixel: x = x + 1.

- Nếu p < 0: p = p + 4x + 6.

Ngược lại: p = p + 4(x - y) + 10 và y = y – 1.

- Vẽ điểm 8 điểm ứng với (x,y) mới.

• **Bước 3**: Lặp lại bước 2 cho đến khi x = y.

1. **Cài đặt thuật toán:**
   1. **Ý tưởng:**

Nhập các thông số cần thiết và tính ∆x, ∆y, m.

Xét các trường hợp :

* Nếu m>1 và m<-1 có điểm chung là thay ∆x, ∆y cho nhau.
* Nếu m<0 thì ∆y = -∆y.

Tính e(Ti), e(Si), e1.

Xét :

* Nếu m>1 hoặc m<-1 lặp điều kiện yA!=yB (y tăng chậm hơn).

o Putpixel(x,y,c).

o Nếu e<0 (ban đầu là e1) thì e+= e(Ti).

o Ngược lại e+=e(Si) và tăng x lên 1 đơn vị

* Nếu m>1 thì tăng y lên 1 và ngược lại giảm y 1 đơn vị.

o Lặp điều kiện xA!=xB (x tăng chậm hơn y).

o Putpixel(x,y,c).

o Nếu e<0 e+=e(Ti).

o Ngược lại e+=e(Si).

o m>0 tăng y.

o m<0 giảm y.

Tăng x lên 1 đơn vị.

* 1. **Cài đặt thuật toán**

#include<conio.h>

#include<stdlib.h>

#include<math.h>

#include<graphics.h>

#include<dos.h>

#include<stdio.h>

void veduongthang(int x1,int y1,int x2,int y2,int mau)

{

int x,y,dx,dy,e,et,ekt,c=mau;

float m;

dx=x2-x1;

dy=y2-y1;

if (dx==0)

{ //outtextxy(100,100,"Truong hop dx==0");

for(x=x1,y=(y1<y2)?y1:y2;(y<=y2||y<=y1);y++)

putpixel(x,y,c);

}

else if (dy==0)

{ //outtextxy(100,100,"Truong hop dy==0");

for(x=(x1<x2)?x1:x2,y=y1;(x<=x2)||(x<=x1);x++)

putpixel(x,y,c);

}

else if (dx<0) veduongthang(x2,y2,x1,y1,mau);

else { m=(float)dy/dx;

if(m==1)

{ // outtextxy(100,100,"Truong hop m==1");

for(x=x1,y=y1;x<=x2;x++,y++)

{

putpixel(x,y,c);

}

}

else if(m==-1)

{ //outtextxy(100,100,"Truong hop m==-1");

for(x=x1,y=y1;x<=x2;x++,y--)

putpixel(x,y,c);

}

else if(m<-1)

{ // outtextxy(100,100,"Truong hop m<-1");

ekt=dx+dx;

e=ekt+dy;

et=e+dy;

for (x=x2,y=y2;y<=y1;y++)

{

putpixel(x,y,c);

if (e<0) e+=ekt;

else

{

x--;

e+=et;

}

}

}

else if((m>-1)&&(m<0))

{ //outtextxy(100,100,"Truong hop -1<m<0");

ekt=-dy-dy;

e=ekt-dx;

et=e-dx;

for (x=x1,y=y1;x<=x2;x++)

{

putpixel(x,y,c);

if (e<0) e+=ekt;

else

{

y--;

e+=et;

}

}

}

else if (m>1)

{

//outtextxy(100,100,"Truong hop m>1");

e=2\*dx-dy;

et=2\*dx-2\*dy;

ekt=2\*dx;

for(x=x1,y=y1;y<=y2;y++)

{

putpixel(x,y,c);

if(e<0) e+=ekt;

else {

x++;

e+=et;

}

}

}

else {

//outtextxy(100,100,"Truong hop 0<m<1");

ekt=dy+dy;

e=ekt-dx;

et=e-dx;

for(x=x1,y=y1;x<=x2;x++)

{

putpixel(x,y,c);

if(e<0) e+=ekt;

else

{

y++;

e+=et;

}

}

}

}

}

main()

{ int i;

initwindow(640,480,"WinBGIm");

//ve may nha

veduongthang(70,150,320,50,4);

veduongthang(320,50,570,150,4);

veduongthang(570,150,70,150,4);

//ve khung nha

rectangle(120,150,520,450);

//ve cua chinh

rectangle(170,300,250,450);

//ve khoa cua chinh

circle(240,380,5);

setfillstyle(1,4);

floodfill(240,380,15);

//ve cua so

rectangle(370,200,450,280);

getch();

closegraph( );

}

**3.1 Minh họa thuật toán**

****

# KẾT LUẬN

Đề tài của nhóm chúng em có phần nào đó giúp mọi người hiểu rõ hơn về thuật toán Bresenham trong Đồ Họa Máy Tính và có thể ứng dụng trong học tập.

Qua việc thực hiện đề tài: **“*Viết chương trình cài đặt thuật toán Bresenham. Ứng dụng thuật toán Bresenham để vẽ ngôi nhà*”**, nhóm chúng em có thêm nhiều hiểu biết hơn về kiến thức Đồ Họa Máy Tính nói chung và Đồ Họa Máy Tính trên C nói riêng .

Trong quá trình thực hiện đề tài, vì còn giới hạn về kiến thức nên một số hạn chế của ba thuật toán xén hình nhóm chưa khắc phục được, và đề tài của nhóm khó có thể tránh khỏi thiếu sót. Nhóm rất mong nhận được ý kiến đóng góp của tất cả mọi người để đề tài của nhóm có thể hoàn thiện hơn.

Chúng em xin chân thành cảm ơn**–** đã tận tình chỉ bảo và giúp đỡ chúng em thực hiện đề tài này.

***Một lần nữa chúng em xin chân thành cảm ơn !***