

ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Vũ Văn Quảng

**BÀI TOÁN XÁC ĐỊNH VỊ TRÍ CỦA MỘT ĐIỂM SO VỚI
ĐA GIÁC VÀ ỨNG DỤNG TRONG BẢN ĐỒ SỐ**

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

Mã số: **60 48 0101**

Thái Nguyên, 9 - 2016

ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Vũ Văn Quảng

**BÀI TOÁN XÁC ĐỊNH VỊ TRÍ CỦA MỘT ĐIỂM SO VỚI
ĐA GIÁC VÀ ỨNG DỤNG TRONG BẢN ĐỒ SỐ**

Chuyên ngành: **Khoa học máy tính**

Mã số: **60 48 0101**

Người hướng dẫn: **PGS.TS Đỗ Trung Tuấn**

Thái Nguyên, 9 - 2016

Thái Nguyên, 9 - 2016

Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi, với sự hướng dẫn khoa học của giáo viên.

Các số liệu, kết quả nêu trong luận văn hoàn toàn là trung thực và chưa từng được ai công bố trong bất kỳ tài liệu nào khác.

Mọi tham khảo trong luận văn được trích dẫn rõ ràng tên tôi, tên công trình, thời gian, địa điểm công bố

Nếu phát hiện gian lận tôi xin hoàn toàn chịu trách nhiệm.

Lời cảm ơn

Để hoàn thành chương trình cao học và viết luận văn này, em đã nhận được sự giúp đỡ và đóng góp nhiệt tình của các thầy cô Trường Công nghệ Thông tin và Truyền thông, Đại học Thái Nguyên.

Trước hết, em xin chân thành cảm ơn các thầy cô trong bộ phận Đào tạo sau đại học, Trường Công nghệ thông tin và Truyền thông, Đại học Thái Nguyên đã tận tình giảng dạy, trang bị cho em những kiến thức quý báu trong suốt những năm học qua.

Xin chân thành cảm ơn gia đình, bạn bè đã nhiệt tình ủng hộ, giúp đỡ, động viên cả về vật chất lẫn tinh thần trong thời gian học tập và nghiên cứu.

Trong quá trình thực hiện luận văn, mặc dù đã rất cố gắng nhưng cũng không tránh khỏi những thiếu sót. Kính mong nhận được sự cảm thông và tận tình chỉ bảo của các thầy cô và các bạn.

Mục lục

Lời cam đoan	iii
Lời cảm ơn.....	iv
Mục lục	v
Danh sách các từ viết tắt.....	viii
Danh mục các hình vẽ, bảng biểu.....	ix
Chương mở đầu	1
Đặt vấn đề.....	1
Đối tượng và phạm vi nghiên cứu	2
Ý nghĩa khoa học của đề tài	2
Chương 1	4
Giới thiệu chung về hình học	4
1.1. Tầm quan trọng của hình học trong toán học	4
1.1.1. Hình học thực tiễn	4
1.1.2. Hình học tiên đề	4
1.1.3. Các số trong hình học	4
1.2. Các yếu tố hình học	4
1.2.1. Điểm	5
1.2.2. Đoạn thẳng	5
1.2.3. Đường	6
1.2.4. Đường cong	8
1.2.5. Mặt phẳng	8
1.3. Tập các vùng.....	8
1.3.1. Tam giác	9
1.3.2. Đa giác.....	12
1.4. Kết luận	15
Chương 2	16
Một số thuật toán hình học và bản đồ	16
2.1. Thuật toán hình học.....	16
2.1.1. Khái niệm về thuật toán và hệ tọa độ	16

2.1.2. Một số thuật toán.....	19
2.2. Tìm kiếm vùng	35
2.2.1. Tìm kiếm vùng đơn hình	35
2.2.2. Các biến thể	36
2.3. Thuật toán Ray Casting	36
2.3.1. Kiểm tra một điểm trong một đa giác trên mặt phẳng tọa độ	36
2.3. Kết luận chương	38
Chương 3	40
Khái niệm bản đồ.....	40
3.1. Bản đồ.....	40
3.1.1. Khái niệm bản đồ	40
3.1.2. Bản đồ địa chính.....	41
3.1.3. Bản đồ số	43
3.1.4. ArcGIS, giải pháp toàn diện cho hệ thống thông tin địa lý.....	43
3.1.5. Quy trình lập bản đồ.....	47
3.2. Ứng dụng trên bản đồ cần xác định điểm thuộc đa giác	51
3.2.1. Ứng dụng trên bản đồ địa chính	51
3.2.2. Ứng dụng trên bản đồ số	52
3.2.3. Ứng dụng trên lãnh hải.....	53
3.2.4. Ứng dụng trên không phận.....	53
3.3. Kiểm tra một điểm thuộc vào đa giác nhờ thuật toán Ray Casting.....	54
3.3.1. Môi trường DEV C.....	54
3.3.2. Chương trình thử nghiệm	55
3.4. Kết luận	57
Kết luận	58
Kết quả đạt được.....	58
Phương hướng tiếp tục	59
Tài liệu tham khảo	60
Tiếng Việt.....	60
Tiếng Anh.....	60
Phụ lục	61

Chương trình kiểm tra một điểm thuộc đa giác, theo thuật toán Ray Casting	61
Chương trình cho thuật toán DDA	63
Chương trình cho thuật toán Bresenham.....	64
Chương trình thuật toán vẽ đường tròn	64
Chương trình vẽ đường tròn bằng thuật toán Bresenham	65
Chương trình thuật toán vẽ đường ellipse	65

Danh sách các từ viết tắt

3D	Ba chiều
3D MAX	Phần mềm đồ họa
ArcGIS	Phần mềm dùng cho GIS
AutoCAD	Phần mềm thiết kế tự động
CGI	Mô phỏng hình ảnh nhờ máy tính
CNTT	Công nghệ Thông tin
CS	Khoa học máy tính
DAE	Differential Algebraic Equation phương trình đại số vi phân
ESRI	Environmental System Research Institute
GIS	Hệ thống thông tin địa lí
HTML	Ngôn ngữ đánh dấu siêu văn bản
IDE	Integrated Development Environment
ODE	Ordinary Differential Equation Phương trình vi phân thường
VR	Hiện thực ảo

Danh mục các hình vẽ, bảng biểu

Hình 1.1. Điểm	5
Hình 1.2. Đoạn thẳng	5
Hình 1.3. Đường thẳng trong mặt phẳng.....	6
Hình 1.4. Tia.....	7
Hình 1.5. Đường parabol, ví dụ về đường cong đơn giản.....	8
Hình 1.6. Trục tâm H của tam giác ABC	9
Hình 1.7. Trọng tâm của tam giác	9
Hình 1.8. Đường tròn ngoại tiếp tam giác	10
Hình 1.9. Đường tròn nội tiếp tam giác.....	10
Hình 1.10. Tam giác đều, cân.....	11
Hình 1.11. Góc của tam giác	12
Hình 1.12. Đa giác lồi	13
Hình 1.13. Đa giác lõm	13
Hình 1.14. Đa giác đơn.....	14
Bảng 1.1. Thuật ngữ	15
Hình 3.1. Hệ tọa độ thực	17
Hình 3.2. Hệ tọa độ trên màn hình	18
Hình 3.3. Hệ tọa độ trên màn hình.	18
Hình 2.1. Xác định điểm, đoạn thẳng.....	19
Hình 2.2. Khoảng cách	20
Hình 2.3. Kiểm tra giao của hai đường d_1, d_2	24
Hình 2.4. Các điểm vẽ gần với điểm muốn vẽ	24
Hình 2.6. Sơ đồ khối thuật toán DDA	25
Hình 2.5. Hai dạng đường thẳng có $0 < m < 1$ và $m > 1$	26
Hình 2.7. Dạng đường thẳng có $0 \leq m \leq 1$	26
Hình 2.8. Sơ đồ khối thuật toán Bresenham.....	27
Hình 2.9. Đường tròn với các điểm đối xứng.	29
Hình 2.10. Đường tròn với điểm $Q(x_{i+1}, y)$ và điểm MidPoint.	29

Hình 2.11. Sơ đồ khối thuật toán MidPoint vẽ đường tròn	30
Hình 2.12. Đường tròn với khoảng cách d_1 và d_2	31
Hình 2.13. Hai dạng của đường gấp khúc.	33
Hình 2.14. Đa giác lồi và đa giác lõm	34
Hình 2.15. Đường thẳng AB và 2 điểm C, D.	34
Hình 2.16. Đa giác lồi có 5 đỉnh.....	35
Hình 2.17. Vùng	35
Hình 2.18. Điểm P nằm trong đa giác ABCDEF, thỏa điều kiện trên.	37
Hình 2.19. ý tưởng cho thuật toán Ray Casting	38
Hình 3.1. ArcGIS, sẽ dùng trong luận văn	44
Hình 3.2. Qui trình lập bản đồ	48
Hình 3.3. Bản đồ địa chính.....	52
Hình 3.4. Bản đồ số hóa trong MapInfo.....	52
Hình 3.5. Ngư dân đánh cá trong khu vực qui định.	53
Hình 3.6. Lập đường bay trên không phận.....	54
Hình 3.7. Chương trình DEV C.....	55
Hình 3.8. Thí dụ chọn các tọa độ trên bản đồ trong ArcGIS	55
Hình 3.9. Dữ liệu đầu vào	56
Hình 3.10. Chương trình trong DEV C	57
Hình 3.11. Kết quả đầu ra.....	57

Chương mở đầu

Đặt vấn đề

Hình học là một lĩnh vực trong toán học nghiên cứu về các mối quan hệ giữa các đối tượng như điểm, đường thẳng, mặt phẳng, không gian cùng các tính chất cơ bản của nó dựa trên các hệ tiên đề. Hệ tiên đề bao gồm các khái niệm nguyên thủy không định nghĩa và các tiên đề (còn được gọi là các định đề) không chứng minh quy định mối quan hệ giữa các khái niệm ấy.

Hệ tiên đề hình học đầu tiên được tập hợp hệ thống và công bố trong tác phẩm Cơ sở của Euclid. Hệ tiên đề này lấy mô hình từ không gian vật lý theo nhận thức của thời đó. Các khái niệm nguyên thủy trong hệ tiên đề này là điểm, đường thẳng và mặt phẳng. Từ ba khái niệm cơ bản này và một số rất ít các tiên đề, Euclid đã xây dựng thành nội dung toàn bộ môn hình học ở phổ thông hiện nay, mà sau này các nhà toán học gọi là hình học Euclid.

Tuy nhiên, các tiên đề/định đề và một số khái niệm do Euclid xây dựng chưa đủ chặt chẽ do chưa có sự hoàn thiện về lý thuyết tập hợp. Sau này David Hilbert đã hoàn chỉnh lại thành một hệ tiên đề chặt chẽ và hoàn chỉnh. Môn hình học thường chia ra hình học phẳng và hình học không gian.

Hình học xuất hiện khá sớm. Hàng ngàn năm trước Công nguyên, con người đã phải đo đạc các thửa ruộng, đong thóc gạo khi thu hoạch, xây dựng những kim tự tháp khổng lồ, xác định một điểm nằm trong hay ngoài một tam giác. Hình học lúc đầu ra đời có ý nghĩa là một khoa học về đo đạc. Nhưng rồi, con người không phải chỉ cần đo đất, mà cần nghiên cứu nhiều điều phức tạp hơn. Tuy nhiên, hình học chỉ trở thành môn khoa học thực sự khi con người nêu lên các tính chất hình học bằng con đường suy diễn chặt chẽ, chứ không phải từ đo đạc trực tiếp.

Môn hình học không những là môn học bắt buộc mà còn ứng dụng trong nhiều môn học khác và trong thực tế cuộc sống cho các lực lượng giáo dục cũng như người sử dụng. Quá trình học môn hình học có thể hiểu và áp dụng trong môn học khác như địa lí xác định lãnh thổ một tỉnh hay một quốc gia

Hiện nay công nghệ thông tin nói chung và môn tin học nói riêng bắt đầu từng bước phát triển và là nhu cầu tất yếu trong giáo dục và đào tạo hiện nay. Một trong những quan tâm, liên quan đến luận văn này là có thể kích chuột lên bản đồ số và xác định xem điểm đó có thuộc lãnh thổ nào.

Trong điều kiện hiện nay để xác định một vùng lãnh thổ một nước hay một địa phương... chính xác là một việc hết sức khó khăn. Tuy nhiên ta coi lãnh thổ đó là một đa giác và chia đa giác đó thành những tam giác (số lượng tam giác càng nhiều thì độ

chính xác của lãnh thổ càng cao từ đó việc xác định một điểm có thuộc lãnh thổ đó hay không trên bản đồ số bằng việc xác định điểm đó có thuộc tam giác nào.

Để thực hiện đề tài này ta cần giải quyết hai bài toán sau:

1. Xác định vùng lãnh thổ bằng cách chia các đa giác (Đa giác lồi hoặc đa giác lõm) và chia đa giác đó thành n tam giác (Tuy nhiên trong khuôn khổ đề tài bản Demo $n = 4, 5, 6$).
2. Xác định một điểm thuộc hay không thuộc một tam giác.

Để thực hiện công việc này, ta có thể áp dụng nhiều phương pháp hiện có:

Qua nhu cầu thực tế và khả năng bản thân tôi nhận thấy mình tham gia nghiên cứu và chọn đề tài “*Bài toán xác định vị trí của một điểm so với đa giác và ứng dụng trong bản đồ số*” nhằm mục đích trau dồi kiến thức đồng thời áp dụng phương pháp dạy học tích cực.

Đối tượng và phạm vi nghiên cứu

Luận văn nhằm vào các đối tượng sau:

- Bản đồ số
- Xác định vùng lãnh thổ bằng các đa giác
- Điểm thuộc, không thuộc tam giác

Hướng nghiên cứu của đề tài là:

- Tìm hiểu các cách xác định vùng lãnh thổ bằng các đa giác
- Tìm hiểu điểm thuộc, không thuộc tam giác
- Thu thập các cách tìm điểm thuộc, không thuộc tam giác
- Phân tích, đánh giá qua từng công cụ hỗ trợ.
- Cài đặt thực nghiệm.

Ý nghĩa khoa học của đề tài

Thông qua luận văn tốt nghiệp, bản thân học viên hiểu sâu hơn về môn hình học và sự tương tác giữa người và máy, từ đó có những thay đổi cho phù hợp để quá trình giáo dục đạt kết quả tốt

Dựa trên thực tế các vấn đề học sinh được áp dụng công nghệ thông tin, và sự tương tác giữa người và máy trong học tập gặp phải. từ đó đề xuất các giải pháp trong công tác dạy và học môn hình học, nâng cao chất lượng đào tạo.

Luận văn có cấu trúc theo các chương.

- Chương 1 đề cập vai trò của môn hình học và thách thức về nghiên cứu xác định vị trí trên bản đồ số;

- Chương 2 trình bày một số khái niệm hình học cơ bản, liên quan đến việc xác định vị trí địa lí trên bản đồ và giới thiệu phần mềm quản lí bản đồ trong hệ thống thông tin địa lí GIS;
- Chương 3 là ứng dụng thử nghiệm của luận văn, thể hiện những cài đặt trên máy, cho phép xác định vị trí trên bản đồ số. kèm theo là một số thuật toán hình học, vừa dùng cho công tác giảng dạy, vừa có thể ứng dụng trên bản đồ số.

Cuối luận văn là phần kết luận, với tổng kết công việc đã tìm hiểu và thử nghiệm. Phần phụ lục là một số chương trình nguồn thử nghiệm của luận văn.

Chương 1

Giới thiệu chung về hình học

1.1. Tầm quan trọng của hình học trong toán học

1.1.1. Hình học thực tiễn

Hình học có nguồn gốc là một khoa học thực tiễn liên quan đến khảo sát, đo đạc, diện tích và khối lượng. Bao gồm các công thức về độ dài, diện tích và thể tích, các phương pháp tính toán các khoảng cách và chiều cao. Sự phát triển của thiên văn học dẫn đến sự ra đời của lượng giác phẳng và lượng giác cầu, cùng với các kỹ thuật tính toán.

1.1.2. Hình học tiên đề

Euclid là người đầu tiên đề xuất về hình học tiên đề, đó là thể hiện tính chất cơ bản hoặc hiển nhiên đúng của điểm, đường thẳng, và mặt phẳng, suy luận một cách chặt chẽ để rút ra các định lý khác bằng cách lý luận toán học. Tính năng đặc trưng của phương pháp tiếp cận của hình học Euclid là sự chặt chẽ của nó, và đã được biết đến như hình học tiên đề hoặc hình học tổng hợp. Vào đầu thế kỷ 19, việc khám phá hình học phi Euclid của Nikolai Ivanovich Lobachevsky (1792–1856), János Bolyai (1802–1860), Carl Friedrich Gauss (1777–1855) và những người khác dẫn đến một sự quan tâm trở lại trong phương pháp tiếp cận này, và trong thế kỷ 20, David Hilbert (1862–1943) đã áp dụng lý luận tiên đề nhằm cung cấp một nền tảng hiện đại của hình học.

1.1.3. Các số trong hình học

Các số đã được giới thiệu trở lại trong hình học dưới hình thức hệ tọa độ của Descartes, người đã nhận ra rằng việc nghiên cứu các hình dạng hình học có thể được hỗ trợ bằng các diễn đạt đại số của chúng. Hình học giải tích ứng dụng các phương pháp của đại số để giải quyết các bài toán hình học, bằng cách liên hệ các đường cong hình học với các phương trình đại số. Những ý tưởng này đóng một vai trò quan trọng trong sự phát triển của vi phân và tích phân trong thế kỷ 17 và đã dẫn đến việc phát hiện ra nhiều đặc tính mới của đường cong phẳng. Hình học đại số hiện đại xem xét những câu hỏi tương tự như trên ở một mức độ trừu tượng cao hơn.

1.2. Các yếu tố hình học

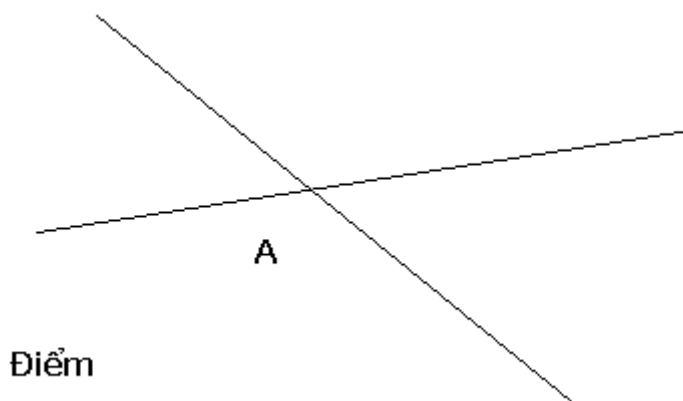
1.2.1. Điểm

Theo [6] trong hình học, điểm là một khái niệm nguyên thủy, không định nghĩa, là cơ sở để xây dựng các khái niệm khác. Điểm được hiểu như là phần của không gian có kích thước mọi chiều bằng không.

Một điểm cũng là một hình. Mỗi đường hình là tập hợp các điểm. Ví dụ: Đường tròn là tập hợp các điểm có cùng bán kính và tâm,...

Nếu điểm nằm trên đường thẳng thì điểm đó chia đường thẳng ra làm 2 phần bằng nhau gọi là tia hay nửa đường thẳng.

Hai điểm giới hạn đoạn thẳng là hai mút của hai đoạn thẳng. Nếu hai đoạn thẳng có chung một điểm, thì $AM + MB = AB$.

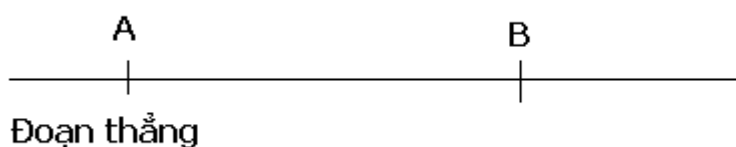


Hình 1.1. Điểm

1.2.2. Đoạn thẳng

Trong hình học, một đoạn thẳng là một phần của đường thẳng mà bị giới hạn bởi hai đầu mút, và là quỹ tích của tất cả những điểm nằm giữa hai đầu mút này trong quan hệ thẳng hàng.

Các ví dụ về đoạn thẳng là: các cạnh của một tam giác hay một hình vuông. Tổng quát hơn, nếu cả hai đầu mút là hai đỉnh kề nhau của một đa giác, đoạn thẳng đó là một cạnh (của đa giác đang xét), nếu hai đầu mút không phải là hai đỉnh kề nhau thì đoạn thẳng đó là đường chéo của đa giác. Khi các đầu mút nằm trên cùng một đường như là đường tròn, thì đoạn thẳng đó được gọi là một dây cung (của đường đang xét).



Hình 1.2. Đoạn thẳng

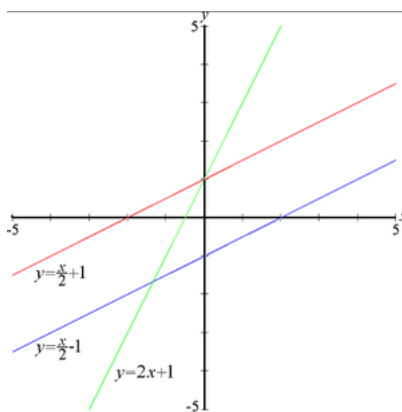
1.2.3. Đường

Đường thẳng là một khái niệm nguyên thủy không định nghĩa, được sử dụng làm cơ sở để xây dựng các khái niệm toán học khác. Đường thẳng được hiểu như cái gì đó không có chiều rộng (không gian một chiều) có độ cong bằng không tại mọi điểm.

Một đường thẳng được hiểu như là một đường dài (vô hạn), mỏng (vô cùng) và thẳng tuyệt đối. Trong hình học Euclide, có một và chỉ có một đường thẳng đi qua hai điểm bất kỳ khác nhau. Đường thẳng này tạo ra đoạn nối ngắn nhất giữa hai điểm đó.

Hai hay ba điểm nằm trên cùng một đường thẳng được gọi là cộng tuyến. Trong một mặt phẳng, hai đường thẳng khác nhau hoặc là song song tức không bao giờ gặp nhau, hoặc giao nhau tại một và chỉ một điểm. Hai mặt phẳng giao nhau nhiều nhất là một đường thẳng.

Đường thẳng trong mặt phẳng Đề các có thể được mô tả bằng phương trình tuyến tính và hàm tuyến tính.



Hình 1.3. Đường thẳng trong mặt phẳng

Khái niệm trực quan về đường thẳng có thể được hình thức hóa bằng nhiều cách. Nếu hình học được phát triển theo phương pháp tiên đề (như trong tác phẩm Các phần tử của Euclid hay trong tác phẩm sau này Cơ sở của hình học của David Hilbert), thì đường thẳng chẳng được định nghĩa gì cả, mà chỉ được đặc trưng bởi các tính chất của nó trong hệ tiên đề. "Bất kỳ thứ gì thỏa mãn các tiên đề của đường thẳng thì nó chính là đường thẳng.". Trong khi Euclide đã từng định nghĩa đường thẳng là cái gì đấy "có chiều dài mà không có bề dày", thực ra ông chưa bao giờ dùng định nghĩa mơ hồ này ở các chứng minh phía sau trong tác phẩm của mình.

Trong không gian Euclide \mathbb{R}^n (và cũng như trong mọi không gian vector khác), chúng ta định nghĩa đường thẳng L là tập con của không gian đang xét và có dạng

$$L = \{\mathbf{a} + t\mathbf{b} \mid t \in \mathbb{R}\}$$

với a và b là hai vector cho trước trong \mathbb{R}^n , đồng thời b phải khác 0. Vector b xác định hướng của đường thẳng, và a là một điểm nằm trên đường thẳng. Chọn các vector a và b khác nhau có thể dẫn đến kết quả cùng một đường thẳng.

Trong không gian hai chiều, chẳng hạn trong một mặt phẳng, hai đường thẳng khác nhau hoặc là hai đường thẳng song song hoặc phải cắt nhau tại một điểm. Tuy nhiên, trong không gian nhiều hơn hai chiều, hai đường thẳng có thể không song song nhau mà cũng chẳng cắt nhau, và hai đường thẳng như vậy gọi là hai đường thẳng chéo nhau.

Trong \mathbb{R}^2 , mọi đường thẳng được biểu diễn bởi một phương trình tuyến tính có dạng

$$L = \{(x, y) \mid ax + by = -c\}$$

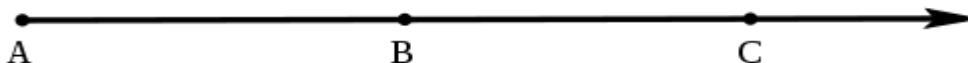
với a , b và c là các hệ số thực cố định trong đó a và b không đồng thời bằng 0 (xem phần phương trình tuyến tính để có thêm các dạng khác). Các tính chất quan trọng của đường thẳng trong không gian hai chiều là độ dốc, giao điểm của nó với trục x , giao điểm của nó với trục y .

Trừu tượng hơn, người ta thường nghĩ về trục số thực như là một nguyên mẫu điển hình cho một đường thẳng, và giả định rằng mỗi điểm trên đường thẳng tương ứng một-một với một số thực nào đó trên trục số thực. Thế nhưng ta hoàn toàn có thể sử dụng cả số siêu thực và kể cả đường thẳng dài trong lý thuyết topo để làm nguyên mẫu cho đường thẳng.

Tính chất "thẳng" của đường thẳng, thường được hiểu là tính chất cho phép đường thẳng cực tiểu hóa khoảng cách giữa hai điểm, mà về sau có thể được tổng quát hóa thành khái niệm đường trắc địa trong đa tạp khả vi.

Phương trình đường thẳng có dạng $y=ax+b$ trong đó a là hệ số góc. Hoặc tổng quát hơn là phương trình $ax+by+c=0$.

Trong hình học Oclít, nếu cho một đường thẳng l và hai điểm A và B , một tia, hay nửa đường thẳng, có gốc A và đi qua B là tập hợp các điểm C trên đường thẳng l sao cho A và B đều thuộc tập hợp này và A không nằm giữa C và B . Điều này có nghĩa là, trong hình học, một tia phát xuất từ một điểm rồi đi mãi về một hướng.

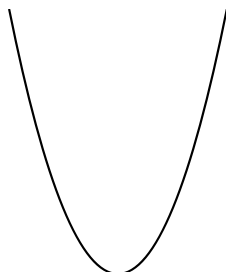


Hình 1.4. Tia

Trong quang học, nhất là trong quang hình, đường lan truyền của ánh sáng hoặc các bức xạ điện từ khác, trong môi trường đồng nhất, là một đường thẳng và được gọi là tia sáng hay quang tuyến. Tia này vuông góc với mặt sóng trong lý thuyết quang sóng.

1.2.4. Đường cong

Trong toán học, đường cong nói tổng quát là một đối tượng tương tự như đường thẳng nhưng không đòi hỏi nó phải thẳng. Điều này nói lên là đường thẳng là một trường hợp đặc biệt của đường cong, hay đường cong có độ cong bằng 0. Các đường cong hai chiều (đường cong phẳng) hoặc đường cong ba chiều trong không gian Euclid là những đối tượng được quan tâm nghiên cứu nhiều.



Hình 1.5. Đường parabol, ví dụ về đường cong đơn giản.

Nhiều bộ môn toán học đã được gán cho các ý nghĩa khác nhau phụ thuộc vào lĩnh vực nghiên cứu, do vậy ý nghĩa chính xác phụ thuộc vào bối cảnh đề cập tới chúng. Tuy thế, các ý nghĩa này là những ví dụ cụ thể của một định nghĩa tổng quát hơn. Chẳng hạn, đường cong là một không gian tô pô mà ánh xạ đồng phôi cục bộ vào một đường thẳng. Trong ngôn ngữ thường ngày, điều này có nghĩa là đường cong là tập hợp các điểm mà tại lân cận đủ nhỏ của mỗi điểm trên nó sẽ nhìn giống như một đường thẳng khi bỏ qua những biến dạng nhỏ. Ví dụ về đường cong như đường parabol bên cạnh. Một số đường cong đặc biệt đã được nghiên cứu trong nhiều lĩnh vực toán học khác nhau.

Các khái niệm liên hệ gần gũi với đường cong đó là "đồ thị của hàm số" (chẳng hạn "đường cong Phillips") và "đồ thị hai chiều hoặc đồ thị ba chiều không có nút thắt".

1.2.5. Mặt phẳng

Mặt phẳng là một khái niệm cơ bản trong toán học (được thừa nhận không định nghĩa), là một tập hợp tất cả các điểm trong không gian ba chiều mà tọa độ Descartes x, y, z của chúng thoả mãn một phương trình có dạng:

$$ax + by + cz + d = 0$$

trong đó a, b, c, d là các hằng số sao cho a, b, c không đồng thời bằng 0, còn vectơ vuông góc là a_i, b_j, c_k .

Mặt phẳng được hình dung chỉ có chiều dọc và chiều ngang mà không có chiều dày.

1.3. Tập các vùng

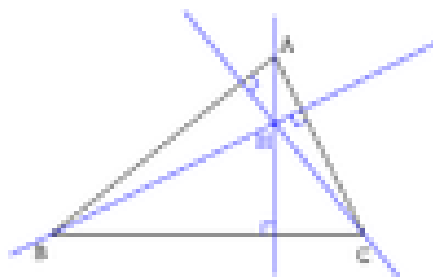
1.3.1. Tam giác

1.3.1.1. Giới thiệu tam giác

Tam giác hay hình tam giác là một loại hình cơ bản trong hình học: hình hai chiều phẳng có ba đỉnh là ba điểm không thẳng hàng và ba cạnh là ba đoạn thẳng. Tam giác là đa giác có số cạnh ít nhất.

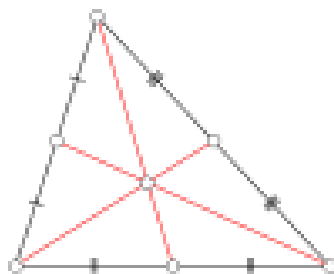
Tam giác luôn luôn là đa giác đơn, lồi. Một tam giác có ba cạnh, ba cạnh ấy tạo thành ba góc, chúng còn được gọi là các góc trong để phân biệt với các góc ngoài là góc kề bù với chúng tạo bởi một cạnh và một cạnh kéo dài.

Trong hình bên A' là góc đối của A đã dịch chuyển, B' là góc đối của B đã dịch chuyển



Hình 1.6. Trực tâm H của tam giác ABC

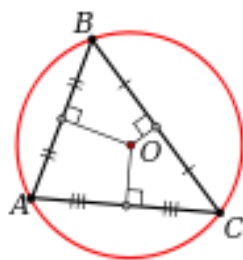
Đoạn thẳng nối một đỉnh với hình chiếu vuông góc của nó trên cạnh đối diện được gọi là đường cao của tam giác. Một tam giác có ba đường cao. Ba đường cao của một tam giác cắt nhau tại một điểm, điểm này được gọi là trực tâm của tam giác



Hình 1.7. Trọng tâm của tam giác

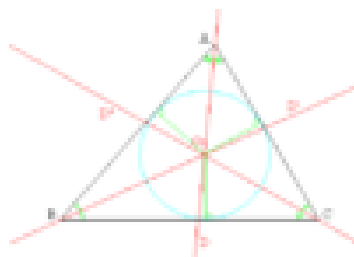
Đoạn thẳng nối mỗi đỉnh với trung điểm của cạnh đối diện được gọi là trung tuyến của tam giác, một tam giác có ba đường trung tuyến. Ba đường trung tuyến của một tam giác cắt nhau tại một điểm, điểm này được gọi là trọng tâm của tam giác.

Trong mặt phẳng, mọi đường thẳng đi qua một đỉnh và trọng tâm của tam giác đều chia tam giác thành hai phần có diện tích bằng nhau



Hình 1.8. Đường tròn ngoại tiếp tam giác

Ngoài ra ba đường trung trực của ba cạnh cắt nhau tại một điểm, đó là tâm đường tròn ngoại tiếp của tam giác.



Hình 1.9. Đường tròn nội tiếp tam giác

Ba đường phân giác của ba góc trong cắt nhau tại một điểm, điểm này là tâm đường tròn nội tiếp tam giác.

Tâm đường tròn nội tiếp tam giác thì cách đều ba cạnh của tam giác

Hai tam giác được gọi là bằng nhau nếu chúng có thể đặt trùng khít lên nhau sau một số phép tịnh tiến, quay và đối xứng. Nói cách khác hai tam giác được gọi là bằng nhau nếu chúng có các cạnh tương ứng bằng nhau và các góc tương ứng bằng nhau. Hai tam giác bằng nhau khi và chỉ khi thỏa mãn một trong ba điều kiện sau:

- Hai tam giác có ba cặp cạnh tương ứng bằng nhau thì bằng nhau (cạnh-cạnh-cạnh).
- Hai tam giác có hai cặp cạnh tương ứng bằng nhau và cặp góc xen giữa các cạnh đó bằng nhau thì bằng nhau (cạnh-góc-cạnh).
- Hai tam giác có một cặp cạnh bằng nhau và hai cặp góc kề với cặp cạnh ấy bằng nhau thì bằng nhau (góc-cạnh-góc).

Trong tam giác đều hai đường tròn nội tiếp và ngoại tiếp đồng tâm với nhau.

Hai tam giác được gọi là đồng dạng nếu một trong chúng bằng với một tam giác nhận được từ tam giác kia sau một phép vị tự. Các điều kiện cần và đủ để hai tam giác đồng dạng:

- Hai tam giác có các cặp cạnh tương ứng tỷ lệ thì đồng dạng.
- Hai tam giác có hai cặp góc tương ứng bằng nhau thì đồng dạng.

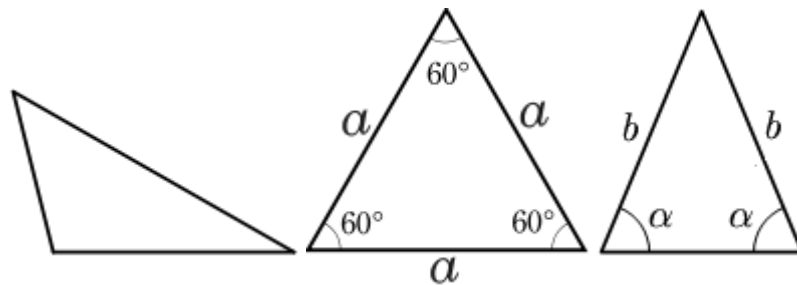
- Hai tam giác có hai cặp cạnh tương ứng tỷ lệ, góc xen giữa hai cặp cạnh ấy bằng nhau thì đồng dạng.

Các tính chất khác:

- Tỷ số hai đường phân giác, hai đường cao, hai đường trung tuyến, hai bán kính nội tiếp và ngoại tiếp, hai chu vi tương ứng của hai tam giác đồng dạng bằng tỷ số đồng dạng.
- Tỷ số diện tích của hai tam giác đồng dạng thì bằng bình phương tỷ số đồng dạng.

Trong hình học Euclid thuật ngữ "tam giác" thường được hiểu là tam giác nằm trong một mặt phẳng. Ngoài ra còn có tam giác cầu trong hình học cầu, tam giác hyperbol trong hình học hyperbol. Tam giác phẳng có một số dạng đặc biệt, xét theo tính chất các cạnh và các góc của nó:

- Trong tam giác thường, mọi cạnh có độ dài khác nhau, mọi góc trong cũng khác nhau.



Tam giác thường

Tam giác đều

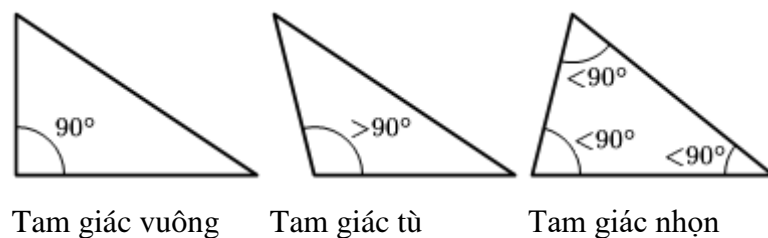
Tam giác cân

Hình 1.10. Tam giác đều, cân

- Tam giác đều là tam giác có cả ba cạnh có độ dài bằng nhau, nói cách khác: ba góc trong bằng nhau và có giá trị bằng $\pi/3$ rad.
- Tam giác cân là tam giác có hai cạnh có độ dài bằng nhau, các cạnh này được gọi là cạnh bên, nói cách khác: tam giác cân là tam giác có hai góc trong bằng nhau (chúng được gọi là các góc ở đáy).

Tam giác vuông là tam giác có một góc bằng $\pi/2$ rad, góc vuông. Trong một tam giác vuông, cạnh đối diện với góc vuông gọi là cạnh huyền, là cạnh lớn nhất. Hai cạnh kia là cạnh góc vuông của tam giác vuông. Định lý Pytago là định lý nổi tiếng đối với hình tam giác vuông, mang tên nhà toán học, triết gia Pytago.

- Tam giác tù là tam giác có một góc trong lớn hơn $\pi/2$ rad (một góc tù).
- Tam giác nhọn là tam giác có ba góc trong đều nhỏ hơn $\pi/2$ rad (ba góc nhọn).



Hình 1.11. Góc của tam giác

Một số tam giác khác là trường hợp đặc biệt trong các phân lớp kể trên. Thí dụ: Tam giác vuông cân vừa là tam giác vuông vừa là tam giác cân.

1.3.1.2. Tính chất của tam giác

Tam giác một vài tính chất sau:

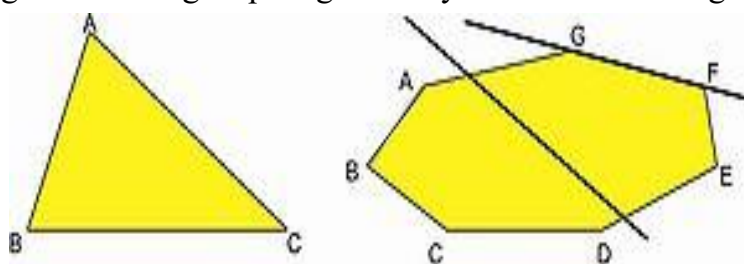
- Tổng các góc trong của một tam giác bằng hai góc vuông (π rad hay 180°).
- Độ dài mỗi cạnh lớn hơn hiệu độ dài hai cạnh kia và nhỏ hơn tổng độ dài của chúng.
- Ba đường cao của tam giác cắt nhau tại một điểm được gọi là trực tâm của tam giác.
- Ba đường trung tuyến của tam giác cắt nhau tại một điểm được gọi là trọng tâm của tam giác. Mọi đường thẳng đi qua trọng tâm của tam giác đều chia tam giác thành hai phần có diện tích bằng nhau.
- Ba đường trung trực của tam giác cắt nhau tại một điểm là tâm đường tròn ngoại tiếp của tam giác.
- Ba đường phân giác trong của tam giác cắt nhau tại một điểm là tâm đường tròn nội tiếp của tam giác.
- Trong hai cạnh của cùng một tam giác cạnh đối diện với góc lớn hơn có chiều dài lớn hơn. Góc đối diện với cạnh lớn hơn là góc lớn hơn.
- Định lý hàm số cosin: Trong một tam giác, bình phương độ dài một cạnh bằng tổng bình phương độ dài hai cạnh còn lại trừ đi hai lần tích của độ dài hai cạnh ấy với cosin của góc xen giữa hai cạnh đó.
- Định lý hàm số sin: Trong một tam giác tỷ lệ giữa độ dài của mỗi cạnh với sin của góc đối diện là như nhau cho cả ba cạnh.

1.3.2. Đa giác

Trong hình học phẳng, đa giác là một đường gấp khúc phẳng khép kín, nghĩa là gồm những đoạn thẳng nối tiếp nhau (mỗi điểm nối là đầu mút của vừa đúng hai đoạn thẳng) cùng nằm trên một mặt phẳng và khép kín (điểm nối đầu trùng với điểm nối cuối). Phần mặt phẳng giới hạn bởi đường đa giác được gọi là hình đa giác.

Những đoạn thẳng trên đường gấp khúc này được gọi là các cạnh của đa giác, còn điểm nối tiếp giữa hai cạnh được gọi là đỉnh của đa giác. Hai cạnh có chung đỉnh cũng được gọi là hai cạnh kề nhau. Nếu đa giác là đa giác đơn thì các cạnh và các đỉnh tạo thành ranh giới của miền đa giác, đôi khi thuật ngữ đa giác nói đến phần trong của đa giác (diện tích mở ở giữa hình này) hay cả miền trong và ranh giới.

Đôi khi người ta cũng xét tới các đường gấp khúc, khép kín, không cùng nằm trong một mặt phẳng, người ta gọi chúng là các đa giác gồnh. Tuy nhiên, thuật ngữ đa giác thường dùng cho các đa giác phẳng. Bài này chỉ nói về các đa giác phẳng.



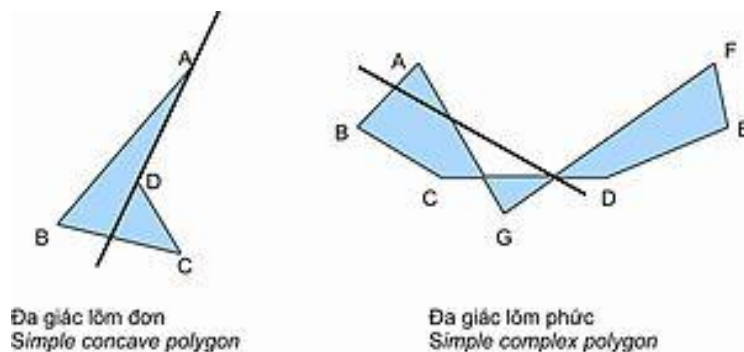
Hình 1.12. Đa giác lõm

1.3.2.1. Đa giác lõm

Đa giác lõm (Convex polygon): toàn bộ đa giác nằm về một phía của đường thẳng chứa cạnh bất kỳ nào của đa giác.

Khi đó, đoạn thẳng nối hai điểm bất kỳ nào của đa giác đều nằm hoàn toàn trong đa giác. Xem thêm liên thông

- Mọi đường thẳng không chứa cạnh đa giác đều chỉ có thể cắt đường đa giác tại nhiều nhất hai điểm. Mọi góc trong đa giác lõm đều không vượt quá 180° . Tổng các góc trong một đa giác lõm n cạnh bằng $(n-2)180^\circ$.
- Đa giác lõm là đa giác đơn.
- Đa giác lõm sao là đa giác có tồn tại điểm x_0 sao cho đoạn thẳng nối x_0 đến điểm bất kỳ y nằm trong đa giác cũng đều được chứa trong đa giác đó

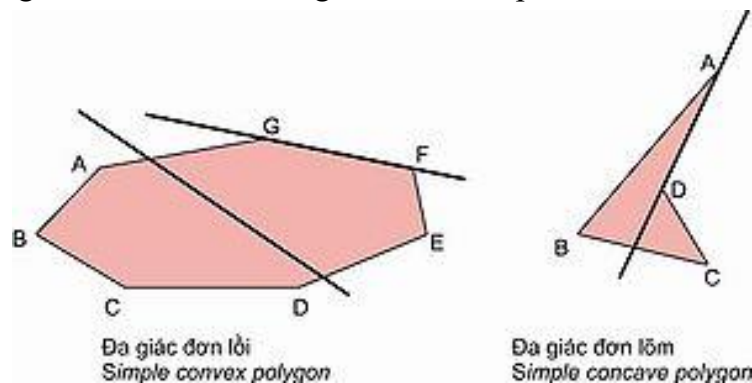


Hình 1.13. Đa giác lõm

1.3.2.2. Đa giác lõm

Đa giác lõm (Concave polygon): đa giác nằm về hai phía của ít nhất một đường thẳng chứa cạnh nào đó.

- Khi đó, có thể có những đoạn thẳng nối hai điểm của đa giác không hoàn toàn nằm trong đa giác, và đường thẳng chứa đoạn thẳng đó cắt đường đa giác tại nhiều hơn hai điểm
- Đa giác lõm nhất định phải có số cạnh lớn hơn hoặc bằng bốn. Tam giác nhất định là đa giác lồi.
- Đa giác lõm có thể là đa giác đơn hoặc phức.



Hình 1.14. Đa giác đơn

1.3.2.3. Đa giác đơn

Đa giác đơn (Simple polygon): đa giác mà các cạnh chỉ có thể cắt nhau tại các đầu mút (đỉnh đa giác), không có hai cạnh không kề nhau cắt nhau.

Đa giác đơn có thể là đa giác lồi hoặc đa giác lõm.

1.3.2.4. Đa giác phức

Đa giác không đơn (đa giác phức-Complex polygon): đa giác có hai cạnh không kề nhau cắt nhau, điểm cắt nhau đó không phải là đỉnh của đa giác.

- Đa giác phức là đa giác lõm;
- Đa giác được gọi là đa giác đều nếu tất cả các cạnh của chúng bằng nhau và tất cả các góc của chúng bằng nhau;
- Đặc biệt tứ giác đều chính là hình vuông;
- Khác với đa diện đều, đa giác đều có thể có số cạnh (góc) lớn vô cùng. Khi đó, hình dáng đa giác đều tiến dần tới hình tròn.

Trong hình học phẳng của một đa giác đơn giản, miền đa giác là tập hợp các điểm trên mặt phẳng "nằm trong" đa giác đơn giản đó.

Đa giác thường được gọi theo số cạnh của nó, người Việt quen dùng các từ chỉ số lượng trong hình học bằng phiên âm Hán-Việt. Ví dụ:

Bảng 1.1. Thuật ngữ

Tên đa giác	tam giác	tứ giác	ngũ giác	lục giác	thất giác	bát giác	thập giác
Số cạnh	3	4	5	6	7	8	10

1.3.2.5. Các loại đa giác khác nhau

Thực ra cách gọi như vậy cũng chỉ có nghĩa là hình ba góc, bốn góc,...Tuy nhiên gần đây có xu hướng Việt hoá các từ này. Trừ các từ tam giác và tứ giác đã quá quen thuộc, người ta đã bắt đầu gọi hình năm cạnh thay cho ngũ giác, hình sáu cạnh thay cho lục giác, hình mười cạnh thay cho thập giác,..., tuy chưa thông dụng lắm. Đặc biệt các đa giác với số cạnh lớn đã thường xuyên được dùng với từ Việt hoá như: hình mười cạnh, hình hai mươi cạnh,... Nếu cần trọng hơn thì dùng từ đa giác mười cạnh, đa giác hai mươi cạnh. Sở dĩ như vậy vì các từ Hán -Việt chỉ số đếm như thập nhất, thập nhị đã dần dần xa lạ với đa số người Việt.

1.4. Kết luận

Chương trên đã đề cập một số khái niệm về hình học, vai trò của hình học đối với phát triển ứng dụng công nghệ thông tin. Các yếu tố hình học cơ bản được trình bày gồm (i) điểm; (ii) đoạn thẳng; (iii) đường thẳng... Các vùng như (i) tam giác; (ii) đa giác... được đề cập.

Các bài toán về hình học phẳng, đặc biệt là bài toán đối với tam giác, có ý nghĩa trong chương trình môn toán phổ thông. Trong chương sau, bài toán về đa giác sẽ đưa ra nhiều ứng dụng hơn nữa trong thực tế.

Nội dung chương trên mô tả các yếu tố hình học, để làm cơ sở cho việc ứng dụng và sử dụng chúng trong bài toán công nghệ thông tin.

Chương 2

Một số thuật toán hình học và bản đồ

2.1. Thuật toán hình học

2.1.1. Khái niệm về thuật toán và hệ tọa độ

2.1.1.1. Thuật toán

Theo [8], thuật toán, còn gọi là giải thuật là một tập hợp hữu hạn của các chỉ thị hay phương cách được định nghĩa rõ ràng cho việc hoàn tất một số sự việc từ một trạng thái ban đầu cho trước; khi các chỉ thị này được áp dụng triệt để thì sẽ dẫn đến kết quả sau cùng như đã dự đoán.

Nói cách khác, thuật toán là một bộ các quy tắc hay quy trình cụ thể nhằm giải quyết một vấn đề trong một số bước hữu hạn, hoặc nhằm cung cấp một kết quả từ một tập hợp của các dữ kiện đưa vào.

Một thuật toán [8] có các tính chất sau:

1. Tính chính xác: để đảm bảo kết quả tính toán hay các thao tác mà máy tính thực hiện được là chính xác.
2. Tính rõ ràng: Thuật toán phải được thể hiện bằng các câu lệnh minh bạch; các câu lệnh được sắp xếp theo thứ tự nhất định.
3. Tính khách quan: Một thuật toán dù được viết bởi nhiều người trên nhiều máy tính vẫn phải cho kết quả như nhau.
4. Tính phổ dụng: Thuật toán không chỉ áp dụng cho một bài toán nhất định mà có thể áp dụng cho một lớp các bài toán có đầu vào tương tự nhau.
5. Tính kết thúc: Thuật toán phải gồm một số hữu hạn các bước tính toán.

2.1.1.2. Hệ tọa độ

Hệ tọa độ thế giới thực, hệ tọa độ thiết bị và hệ tọa độ chuẩn

Một hệ phần mềm đồ họa được mô tả bao gồm 3 miền như sau:

- *Miền điều khiển*: bao bọc toàn bộ hệ thống.
- *Miền thực*: nằm trong miền điều khiển. Khi một số nào đó thâm nhập vào miền thực, nó sẽ được chuyển thành số thực dấu phẩy động, và khi có một số rời khỏi miền này thì nó sẽ được chuyển thành số nguyên có dấu 16 bits.

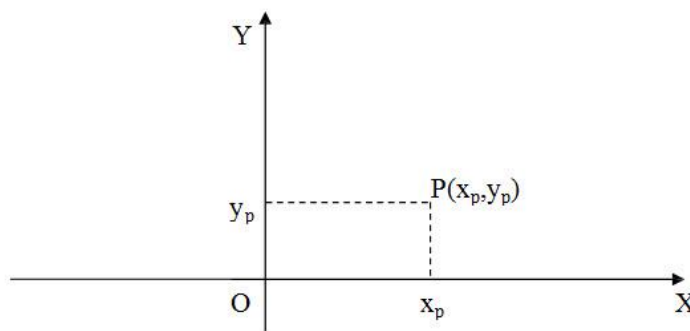
- *Miền hiển thị*: nằm trong miền điều khiển nhưng phân biệt với miền thực. Chỉ có số nguyên 16 bits mới nằm trong miền hiển thị.

Trong lĩnh vực kỹ thuật đồ họa, phải hiểu được rằng thực chất của đồ họa là làm thế nào để có thể mô tả và biến đổi được các đối tượng trong thế giới thực trên máy tính. Bởi vì, các đối tượng trong thế giới thực được mô tả bằng tọa độ thực. Trong khi đó, hệ tọa độ thiết bị lại sử dụng hệ tọa độ nguyên để hiển thị các hình ảnh. Đây chính là vấn đề cơ bản cần giải quyết. Ngoài ra, còn có một khó khăn khác nữa là với các thiết bị khác nhau thì có các định nghĩa khác nhau. Do đó, cần có một phương pháp chuyển đổi tương ứng giữa các hệ tọa độ và đối tượng phải được định nghĩa bởi các thành phần đơn giản như thế nào để có thể mô tả gần đúng với hình ảnh thực bên ngoài.

Hai mô hình cơ bản của ứng dụng đồ họa là dựa trên mẫu số hóa và dựa trên đặc trưng hình học. Trong ứng dụng đồ họa dựa trên mẫu số hóa thì các đối tượng đồ họa được tạo ra bởi lưới các pixel rời rạc. Các pixel này có thể được tạo ra bằng các chương trình vẽ, máy quét,... Các pixel này mô tả tọa độ xác định vị trí và giá trị mẫu. Thuận lợi của ứng dụng này là dễ dàng thay đổi ảnh bằng cách thay đổi màu sắc hay vị trí của các pixel, hoặc di chuyển vùng ảnh từ nơi này sang nơi khác. Tuy nhiên, điều bất lợi là không thể xem xét đối tượng từ các góc nhìn khác nhau. Ứng dụng đồ họa dựa trên đặc trưng hình học bao gồm các đối tượng đồ họa cơ sở như đoạn thẳng, đa giác,... Chúng được lưu trữ bằng các mô hình và các thuộc tính. Ví dụ: đoạn thẳng được mô hình bằng hai điểm đầu và cuối, có thuộc tính như màu sắc, độ dày. Người sử dụng không thao tác trực tiếp trên các pixel mà thao tác trên các thành phần hình học của đối tượng.

Hệ tọa độ thế giới thực

Một trong những hệ tọa độ thực thường được dùng để mô tả các đối tượng trong thế giới thực là hệ tọa độ Descartes. Với hệ tọa độ này, mỗi điểm P được biểu diễn bằng một cặp tọa độ (x_p, y_p) với $x_p, y_p \in \mathbb{R}$ (xem hình 1).



Hình 3.1. Hệ tọa độ thực

Ox: gọi là trục hoành; Oy: gọi là trục tung; x_p : hoành độ điểm P; y_p : tung độ điểm P.

Hệ tọa độ thiết bị

Hệ tọa độ thiết bị (device coordinates) được dùng cho một thiết bị xuất cụ thể nào đó, ví dụ như máy in, màn hình,..

Trong hệ tọa độ thiết bị thì các điểm cũng được mô tả bởi cặp tọa độ (x,y). Tuy nhiên, khác với hệ tọa độ thực là $x, y \in \mathbb{N}$. Điều này có nghĩa là các điểm trong hệ tọa độ thực được định nghĩa liên tục, còn các điểm trong hệ tọa độ thiết bị là rời rạc. Ngoài ra, các tọa độ x, y của hệ tọa độ thiết bị chỉ biểu diễn được trong một giới hạn nào đó của \mathbb{N} .

Độ phân giải của màn hình trong chế độ đồ họa là 640x480. Khi đó, $x \in (0,640)$ và $y \in (0,480)$.



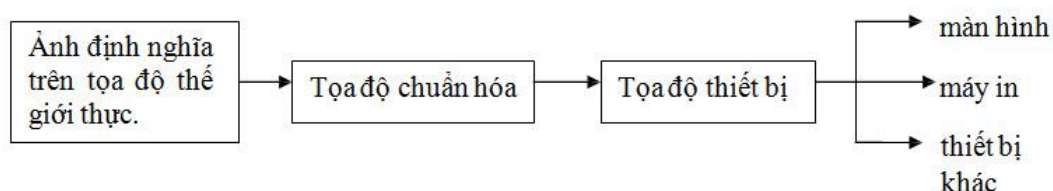
Hình 3.2. Hệ tọa độ trên màn hình

Hệ tọa độ thiết bị chuẩn

Do cách định nghĩa các hệ tọa độ thiết bị khác nhau nên một hình ảnh hiển thị được trên thiết bị này là chính xác thì chưa chắc hiển thị chính xác trên thiết bị khác. Người ta xây dựng một hệ tọa độ thiết bị chuẩn đại diện chung cho tất cả các thiết bị để có thể mô tả các hình ảnh mà không phụ thuộc vào bất kỳ thiết bị nào.

Trong hệ tọa độ chuẩn, các tọa độ x, y sẽ được gán các giá trị trong đoạn từ $[0,1]$. Như vậy, vùng không gian của hệ tọa độ chuẩn chính là hình vuông đơn vị có góc trái dưới $(0, 0)$ và góc phải trên là $(1, 1)$.

Quá trình mô tả các đối tượng thực như sau (xem hình 3):



Hình 3.3. Hệ tọa độ trên màn hình.

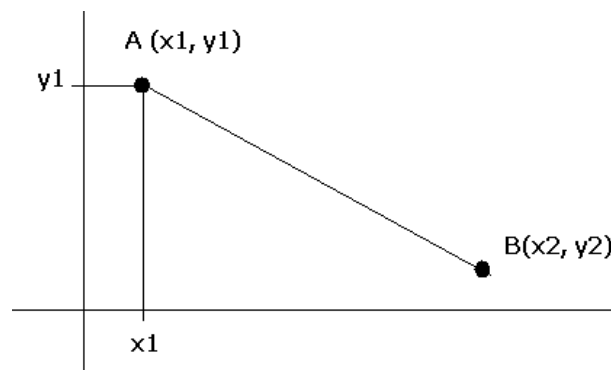
Hình học đối với giác quan của con người thì khá quen thuộc và dễ dàng. Nhưng hình học đối với máy tính thì lại là một vấn đề khác. Nhiều bài toán ta có thể giải ngay lập tức bằng cách “nhìn vào hình vẽ ta thấy”, nhưng để thể hiện trên máy tính thì cần những chương trình không đơn giản chút nào.

2.1.2. Một số thuật toán

2.1.2.1. Biểu diễn yếu tố hình học

Một điểm được biểu diễn trên mặt phẳng nhờ tọa độ Decac. Thí dụ A (x, y) cho phép xác định điểm A.

Đoạn thẳng nối điểm A với điểm B được xác định theo A(x₁, y₁) và B(x₂, y₂).



Hình 2.1. Xác định điểm, đoạn thẳng

Đường thẳng xác định qua phương trình các điểm A thỏa mãn $aX + bY = c$; trong đó a, b, c là số thực. Các điểm A (X, Y) nằm trên đường thẳng, khi X, Y thỏa mãn phương trình trên.

2.1.2.2. Thuật toán tính khoảng cách hai điểm

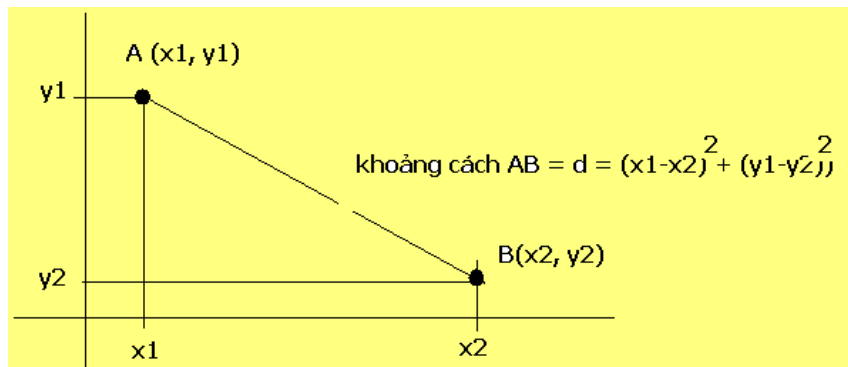
* **Đầu vào** : Hai điểm A, B xác định theo A(x₁, y₁) và B(x₂, y₂).

* **Đầu ra** : Giá trị d.

* **Thuật toán** :

B1- Nhập tọa độ 2 điểm A,B

B2- Khoảng cách AB là $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$



Hình 2.2. Khoảng cách

2.1.2.3. Kiểm tra hai đoạn thẳng cắt nhau

* Đầu vào :

- $A(x_1, y_1), B(x_2, y_2)$ xác định đoạn thẳng d_1
- $C(x_3, y_3), D(x_4, y_4)$ xác định đoạn thẳng d_2 ;

* Đầu ra :

- 1, nếu d_1 cắt d_2
- 0 nếu d_1 không cắt d_2 .

Để giải bài toán này ta nhớ lại một kiến thức đã được học ở phổ thông: trong mặt phẳng tọa độ cho đường thẳng $d: ax + by + c = 0$. Khi ấy, một trong hai nửa mặt phẳng bờ d , ngoại trừ d , gồm các điểm có tọa độ thỏa mãn bất phương trình: $ax + by + c > 0$. Nửa mặt phẳng kia gồm các điểm có tọa độ thỏa mãn bất phương trình: $ax + by + c < 0$.

Từ đó ta dễ dàng chứng minh được kết quả sau: Điều kiện cần và đủ để hai đoạn thẳng AB và CD cắt nhau là: A, B nằm khác phía so với đoạn CD và C, D nằm khác phía so với đoạn AB .

Ngoài ra, cũng không được quên một trong những kiến thức cơ bản là: phương trình đường thẳng đi qua hai điểm phân biệt $A(x_1, y_1), B(x_2, y_2)$ có dạng:

$$f(x,y) = (x-x_1)(y_2-y_1) - (y-y_1)(x_2-x_1) = 0 \quad (d_1)$$

Vậy nếu xét hai điểm $M(x_3, y_3), N(x_4, y_4)$ thì điều kiện cần và đủ để M, N nằm cùng phía (khác phía) so với đường thẳng có phương trình $f(x,y) = 0$ là:

$$f(x_3,y_3) * f(x_4, y_4) >= 0 \quad (<= 0).$$

* Thuật toán :

- B1- Đọc giá trị xác định các đoạn thẳng d_1, d_2 , tức là đọc $(x_i, y_i) \quad i=1..4$;
 - B2- Kiểm tra A, B khác phía đối với đoạn CD ;
 - B3- Kiểm tra C, D khác phía đối với đoạn AB ;
- Nếu các điều kiện trên thỏa mãn, thông báo d_1 cắt d_2 ;

- Ngược lại thông báo d_1 không cắt d_2 .

B4- Kết luận;

Dựa trên cơ sở đó chương trình giải bài toán trên được cài đặt như sau:

Program kiemtra;

Type Diem = Record

 X,y: Real;

 End;

Var A,B,C,D: Diem;

Function F(A,B,M: Diem): Real;

{ Tính giá trị hàm F tại điểm M với F là hàm ứng với
phương trình đường thẳng đi qua AB }

Begin

F := (M.x - A.x) * (B.y - A.y) - (M.y - A.y) * (B.x - A.x);

End;

Function KhacPhia (A,B,C,D:Diem):Boolean;

{ Kiểm tra xem C,D có nằm khác phía so với đường thẳng
đi qua A, B hay không? }

Begin

If F(A,B,C) * F(A,B,D) <= 0 then KhacPhia := True

else khacphia := False;

End;

BEGIN {chương trình chính}

Readln(A.x,A.y,B.x,B.y,C.x,C.y,D.x,D.y);

If Khacphia (A,B,C,D) and Khacphia (C,D,A,B) then

 Writeln ('Hai đoạn thẳng cắt nhau')

else Writeln ('Không cắt nhau');

Readln;

END.

2.1.2.4. Kiểm tra đoạn thẳng cắt đường thẳng

* Đầu vào :

- $A(x_1, y_1), B(x_2, y_2)$ xác định trên đường thẳng d_1
- $C(x_3, y_3), D(x_4, y_4)$ xác định đoạn thẳng d_2 ;

* Đầu ra :

- 1, nếu d_1 cắt d_2
- 0 nếu d_1 không cắt d_2 .

* Thuật toán :

- B1- Nhập tọa độ các điểm A,B,C, D; xác định đường thẳng d_1, d_2 ;
- B2- Xác định C, D cùng phía hay khác phía với d_1 ;
- B3- Xác định A, B cùng phía hay khác phía với d_2 ;
- B4- Kết luận;

Điều kiện cần và đủ để hai đường thẳng AB và đoạn thẳng CD cắt nhau là: C, D nằm khác phía so với đường thẳng AB.

Ngoài ra, cũng không được quên một trong những kiến thức cơ bản là: phương trình đường thẳng đi qua hai điểm phân biệt $A(x_1, y_1), B(x_2, y_2)$ có dạng:

$$f(x,y) = (x-x_1)(y_2-y_1) - (y-y_1)(x_2-x_1) = 0.$$

Vậy nếu xét hai điểm $M(x_3, y_3), N(x_4, y_4)$ thì điều kiện cần và đủ để M, N nằm cùng phía (khác phía) so với đường thẳng có phương trình $f(x,y) = 0$ là:

$$f(x_3,y_3) * f(x_4, y_4) >= 0 (<= 0).$$

Dựa trên cơ sở đó chương trình giải bài toán trên được cài đặt như sau:

Program kiemtra;

Type Diem = Record

 X,y: Real;

End;

Var A,B,C,D: Diem;

Function F(A,B,M: Diem): Real;

{Tính giá trị hàm F tại điểm M với F là hàm ứng với
phương trình đường thẳng đi qua AB }

Begin

F = (M.x - A.x) * (B.y - A.y) - (M.y - A.y) * (B.x - A.x);

End;

Function KhacPhia (A,B,C,D:Diem):Boolean;

{ Kiểm tra xem C,D có nằm khác phía so với đường thẳng đi qua A, B hay không? }

Begin

If $F(A,B,C) * F(A,B,D) <= 0$ then KhacPhia: = True

else khacphia: = False;

End;

BEGIN {chương trình chính}

Readln(A.x,A.y,B.x,B.y,C.x,C.y,D.x,D.y);

If Khacphia (A,B,C,D) then

Writeln ('Hai đoạn thẳng cắt nhau')

else Writeln ('Không cắt nhau');

Readln;

END.

Có thể viết lại thuật toán theo các bước :

1. Đọc giá trị xác định các đoạn thẳng d_1, d_2 , tức là đọc $(x_i, y_i) i=1..4$;
2. Kiểm tra A, B khác phía đối với đoạn CD;
3. Kiểm tra C, D khác phía đối với đoạn AB;
 - Nếu các điều kiện trên thỏa mãn, thông báo d_1 cắt d_2 ;
 - Ngược lại thông báo d_1 không cắt d_2 .

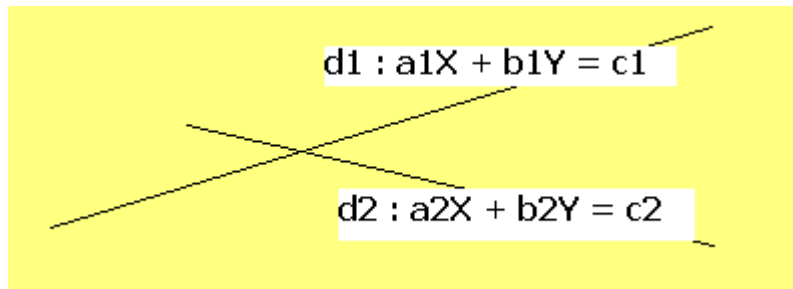
2.1.2.5. Thuật toán kiểm tra hai đường thẳng cắt nhau

* Đầu vào :

- Phương trình đường thẳng $d_1 : a_1x + b_1y = c_1$;
- Phương trình đường thẳng $d_2 : a_2x + b_2y = c_2$.

* Đầu ra

- Thông báo “cắt” nếu d_1 cắt d_2 ;
- Thông báo “không cắt”, nếu ngược lại.



Hình 2.3. Kiểm tra giao của hai đường d1, d2

* Thuật toán :

B1- Tính định thức D: $= a_1 \cdot b_2 - a_2 \cdot b_1$;

B2- Tính D_x : $= c_1 \cdot b_2 - c_2 \cdot b_1$;

B3- Tính D_y : $= a_1 \cdot c_2 - a_2 \cdot c_1$;

B4- Kiểm tra D:

- Nếu $D < 0$ thì thông báo “cắt”;
- Ngược lại thông báo “không cắt”.

2.1.2.5. Thuật toán vẽ đoạn thẳng

Xét đoạn thẳng có hệ số góc $0 < m \leq 1$ và $x > 0$. Với các đoạn thẳng dạng này, nếu (x_i, y_i) là điểm đã được xác định ở bước thứ i thì điểm kế tiếp (x_{i+1}, y_{i+1}) ở bước thứ $i+1$ sẽ là một trong hai điểm sau (xem hình4):

$$\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} = \begin{cases} y_i + 1 \\ y_i \end{cases} \end{cases}$$

Hình 2.4. Các điểm vẽ gần với điểm muốn vẽ

Vấn đề đặt ra là chọn điểm vẽ như thế nào để đường thẳng được vẽ gần với đường thẳng muốn vẽ nhất và đạt được tối ưu hóa về mặt tốc độ ?

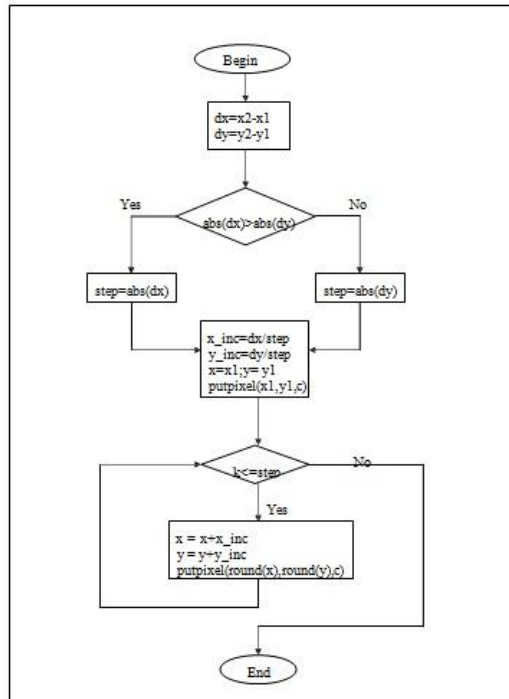
* Đầu vào :

- Cho tọa độ điểm A (X_1, Y_1) ;
- Cho tọa độ điểm B (X_n, Y_n) ;

* Đầu ra :

- Đoạn thẳng AB.

*** Thuật toán DDA (Digital Differential Analyzer)**



Hình 2.6. Sơ đồ khối thuật toán DDA

Là thuật toán tính toán các điểm vẽ dọc theo đường thẳng dựa vào hệ số góc của phương trình đường thẳng $y=mx+b$; trong đó: $m= \Delta x/\Delta y$, $\Delta y=y_{i+1} - y_i$, $\Delta x=x_{i+1} - x_i$

Nhận thấy trong hình vẽ, tọa độ của điểm x sẽ tăng 1 đơn vị trên mỗi điểm vẽ, còn việc quyết định chọn y_{i+1} là y_{i+1} hay y_i sẽ phụ thuộc vào giá trị sau khi làm tròn của tung độ y. Tuy nhiên, nếu tính trực tiếp giá trị thực của y ở mỗi bước từ phương trình $y=mx+b$ thì cần một phép toán nhân và một phép toán cộng số thực.

$$y_{i+1} = mx_i + 1 + b = m(x_i + 1) + b = mx_i + b + m$$

Để cải thiện tốc độ, người ta khử phép nhân trên số thực. Ta có: $y_i = mx_i + b$

$$y_{i+1} = y_i + m \rightarrow \text{int}(y_{i+1})$$

Tóm lại khi $0 < m \leq 1$:

$$\begin{aligned} x_{i+1} &= x_i + 1 \\ y_{i+1} &= y_i + m \rightarrow \text{int}(y_{i+1}) \end{aligned}$$

Trường hợp $m > 1$: chọn bước tăng trên trục y một đơn vị.

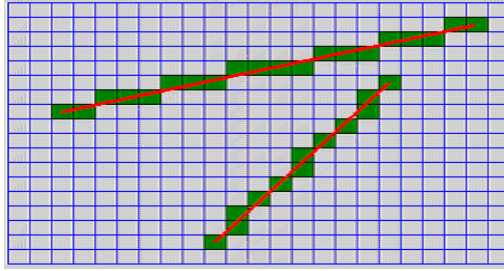
$$\begin{aligned} x_{i+1} &= x_i + 1/m \rightarrow \text{int}(x_{i+1}) \\ y_{i+1} &= y_i + 1 \end{aligned}$$

Hai trường hợp này dùng để vẽ một điểm bắt đầu từ bên trái đến điểm cuối cùng bên phải của đường thẳng (xem hình 5). Nếu điểm bắt đầu từ bên phải đến điểm cuối cùng bên trái thì xét ngược lại:

$$\begin{aligned} 0 < m \leq 1: & \quad x_{i+1} := x_i - 1 \\ & \quad y_{i+1} := y_i - m \rightarrow \text{int}(y_{i+1}) \end{aligned}$$

$$m > 1: x_{i+1} := x_i - 1/m \rightarrow \text{int}(x_{i+1})$$

$$y_{i+1} := y_i - 1$$

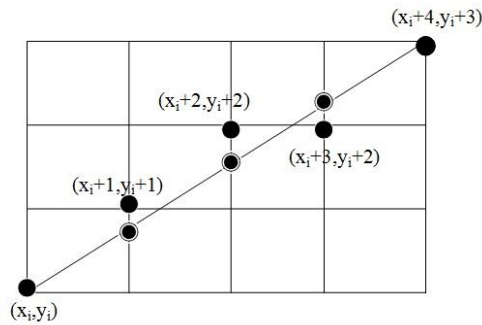


Hình 2.5. Hai dạng đường thẳng có $0 < m < 1$ và $m > 1$

Tương tự, có thể tính toán các điểm vẽ cho trường hợp $m < 0$: khi $|m| \leq 1$ hoặc $|m| > 1$.

Cài đặt minh họa thuật toán DDA trình bày trong phần phụ lục của luận văn.

*** Thuật toán Bresenham**



Hình 2.7. Dạng đường thẳng có $0 \leq m \leq 1$.

B1- Gọi (x_{i+1}, y_{i+1}) là điểm thuộc đoạn thẳng (xem hình 1.6). Ta có $y := m(x_{i+1}) + b$.

$$\text{Đặt } d_1 = y_{i+1} - Y_i$$

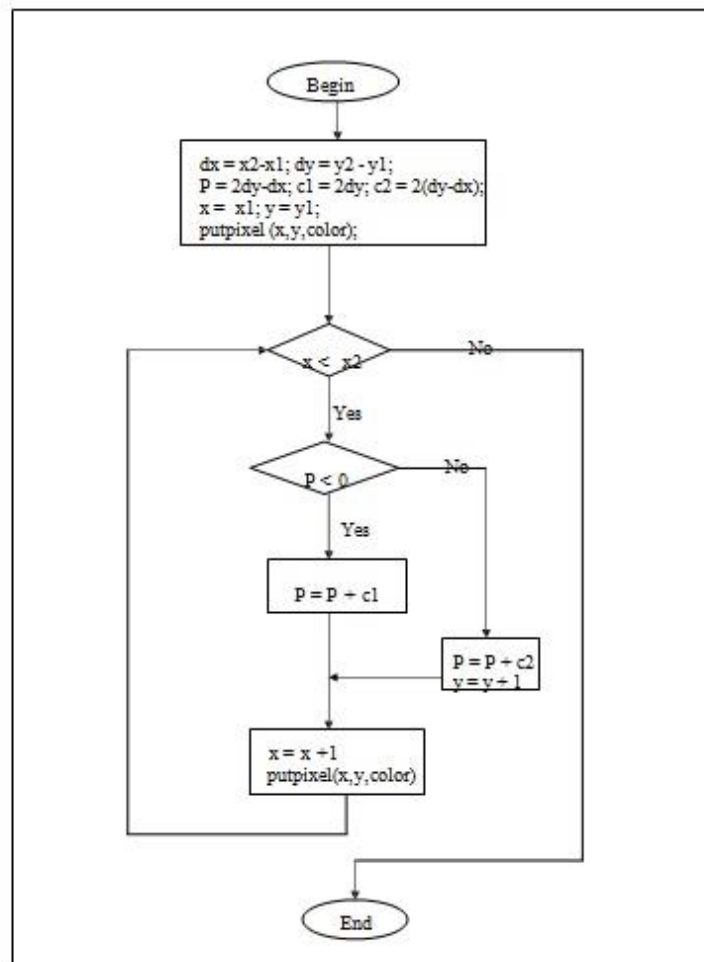
$$d_2 = (Y_{i+1}) - Y_{i+1}$$

Việc chọn điểm (x_{i+1}, y_{i+1}) là P_1 hay P_2 phụ thuộc vào việc so sánh d_1 và d_2 hay dấu của $d_1 - d_2$:

B2

- Nếu $d_1 - d_2 < 0$: chọn điểm P_1 , tức là $y_{i+1} = y_i$
- Nếu $d_1 - d_2 \geq 0$: chọn điểm P_2 , tức là $y_{i+1} = y_{i+1}$

B3- Xét $P_i = \Delta x(d_1 - d_2)$



Hình 2.8. Sơ đồ khối thuật toán Bresenham

Ta có: $d_1 - d_2 = 2y_{i+1} - 2y_{i-1}$

$$= 2m(x_{i+1}) + 2b - 2y_{i-1}$$

$$\Rightarrow P_i = x(d_1 - d_2) = x[2m(x_{i+1}) + 2b - 2y_{i-1}]$$

$$= x[2\Delta y/\Delta x(x_{i+1}) + 2b - 2y_{i-1}]$$

$$= 2y(x_{i+1}) - 2x.y_{i-1} + x(2b - 1)$$

$$= 2y.x_i - 2x.y_i + 2y + x(2b - 1)$$

$$\text{Vậy } C = 2y + x(2b - 1) = \text{Const}$$

$$\Rightarrow P_i = 2y.x_i - 2x.y_i + C$$

B4- Nhận xét rằng nếu tại bước thứ i ta xác định được dấu của P_i thì xem như ta xác định được điểm cần chọn ở bước $(i+1)$. Ta có:

$$P_{i+1} - P_i = (2y.x_{i+1} - 2x.y_{i+1} + C) - (2y.x_i - 2x.y_i + C)$$

$$\Leftrightarrow P_{i+1} = P_i + 2y - 2x(y_{i+1} - y_i)$$

- Nếu $P_i < 0$: chọn điểm P_1 , tức là $y_{i+1} = y_i$ và $P_{i+1} = P_i + 2y$.

- Nếu $P_i \geq 0$: chọn điểm P_2 , tức là $y_{i+1} = y_i + 1$ và $P_{i+1} = P_i + 2y - 2x$

- Giá trị P_0 được tính từ điểm vẽ đầu tiên (x_0, y_0) theo công thức:

$$P_0 = 2y \cdot x_0 - 2x \cdot y_0 + C$$

B5- $D_0(x_0, y_0)$ là điểm nguyên thuộc về đoạn thẳng nên ta có:

$$y_0 = m \cdot x_0 + \Delta y / \Delta x \cdot x_0 + b$$

Thế vào phương trình trên ta được: $P_0 = 2y - x$

Cài đặt minh họa thuật toán Bresenham trình bày trong phụ lục luận văn.

Nhận xét

Thuật toán Bresenham chỉ thao tác trên số nguyên và chỉ tính toán trên phép cộng và phép nhân 2 (phép dịch bit). Điều này là một cải tiến làm tăng tốc độ đáng kể so với thuật toán DDA.

Ý tưởng chính của thuật toán này là ở chỗ xét dấu P_i để quyết định điểm kế tiếp, và sử dụng công thức truy hồi $P_{i+1} - P_i$ để tính P_i bằng các phép toán đơn giản trên số nguyên

Tuy nhiên, việc xây dựng trường hợp tổng quát cho thuật toán Bresenham có phức tạp hơn thuật toán DDA.

2.1.2.6. Thuật toán vẽ đường tròn

* **Đầu vào :**

- Cho tọa độ điểm C (X_c, Y_c); Bán kính R

* **Đầu ra :**

- Đường tròn .

* **Thuật toán** đơn giản: cho $x = 0, 1, 2, \dots, \text{int}((R\sqrt{2})/2)$

- Tại mỗi giá trị x, tính $\text{int}(y = \sqrt{R^2 - x^2})$
- Vẽ điểm (x,y) cùng 7 điểm đối xứng của nó.

Trong hệ tọa độ Descartes, phương trình đường tròn bán kính R có dạng: Với tâm O(0,0): $x^2 + y^2 = R^2$

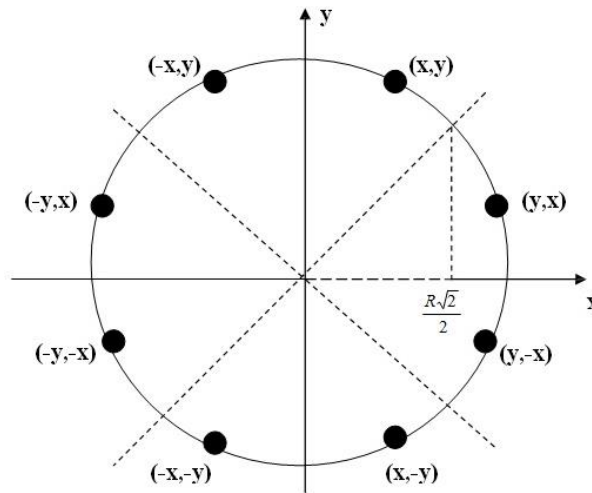
Với tâm C(x_c, y_c): $(x - x_c)^2 + (y - y_c)^2 = R^2$

Trong hệ tọa độ cực: x

$$\begin{cases} x = x_c + R \cdot \cos\theta \\ y = y_c + R \cdot \sin\theta \end{cases}$$

với $\theta \in [0, 2\pi]$.

Do tính đối xứng của đường tròn C (xem hình 9) nên ta chỉ cần vẽ 1/8 cung tròn, sau đó lấy đối xứng qua 2 trục tọa độ và 2 đường phân giác thì ta vẽ được cả đường tròn.



Hình 2.9. Đường tròn với các điểm đối xứng.

Cài đặt minh họa thuật toán đơn giản trong phần phụ lục của luận văn.

Thuật toán xét điểm giữa (MidPoint)

* Đầu vào :

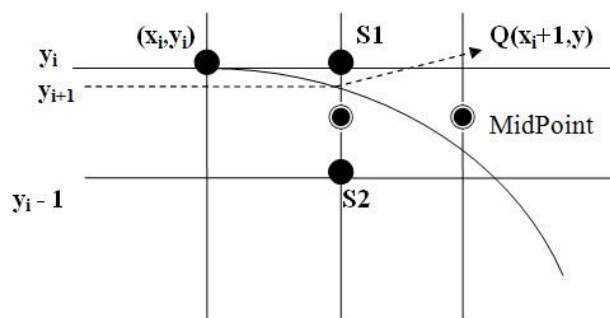
- Cho tọa độ điểm $Q (X_q, Y_q)$; R

* Đầu ra :

- Cung tròn .

* Thuật toán

Do tính đối xứng của đường tròn nên ta chỉ cần vẽ 1/8 cung tròn, sau đó lấy đối xứng là vẽ được cả đường tròn. Thuật toán MidPoint đưa ra cách chọn y_{i+1} là y_i hay y_{i-1} bằng cách so sánh điểm thực $Q(x_{i+1}, y)$ với điểm giữa MidPoint là trung điểm của S_1 và S_2 . Chọn điểm bắt đầu để vẽ là $(0, R)$. Giả sử (x_i, y_i) là điểm nguyên đã tìm được ở bước thứ i (xem hình 10), thì điểm (x_{i+1}, y_{i+1}) ở bước $i+1$ là sự lựa chọn giữa S_1 và S_2 .



Hình 2.10. Đường tròn với điểm $Q(x_{i+1}, y)$ và điểm MidPoint.

Đặt $F(x, y) = x^2 + y^2 - R^2$, ta có:

- $F(x,y) < 0$, nếu điểm (x,y) nằm trong đường tròn.
- $F(x,y) = 0$, nếu điểm (x,y) nằm trên đường tròn.
- $F(x,y) > 0$, nếu điểm (x,y) nằm ngoài đường tròn. Xét $P_i = F(\text{MidPoint}) = F(x_{i+1}, y_{i-1}/2)$. Ta có:

Nếu $P_i < 0$: điểm MidPoint nằm trong đường tròn. Khi đó, điểm thực Q gần với điểm S_1 hơn nên ta chọn $y_{i+1} = y_i$

Nếu $P_i \geq 0$: điểm MidPoint nằm ngoài đường tròn. Khi đó, điểm thực Q gần với điểm S_2 hơn nên ta chọn $y_{i+1} = y_{i-1}$. Mặt khác:

$$\begin{aligned}
 P_{i+1} - P_i &= F(x_{i+1} + 1, y_{i+1} - 1/2) - F(x_i + 1, y_i - 1/2) \\
 &= [(x_{i+1} + 1)^2 + (y_{i+1} - 1/2)^2 - R^2] - [(x_i + 1)^2 + (y_i - 1/2)^2 - R^2] \\
 &= 2x_i + 3 + ((y_{i+1})^2 + (y_i)^2) - (y_{i+1} - y_i)
 \end{aligned}$$

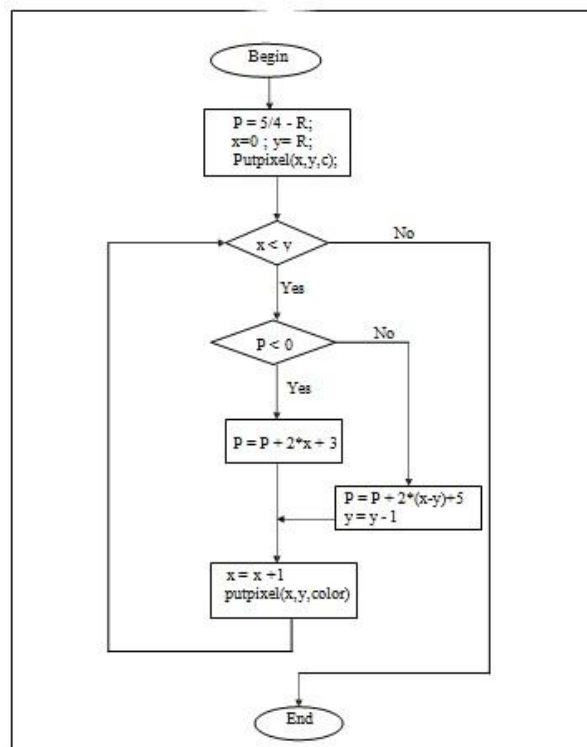
Vậy

Nếu $P_i < 0$: chọn $y_{i+1} = y_i$. Khi đó $P_{i+1} = P_i + 2x_i + 3$

$P_i \geq 0$: chọn $y_{i+1} = y_{i-1}$. Khi đó $P_{i+1} = P_i + 2x_i - 2y_i + 5$.

P_i ứng với điểm ban đầu $(x_0, y_0) = (0, R)$ là:

$$P_0 = F(x_0 + 1, y_0 - 1/2) = F(1, R - 1/2) = 5/4 - R$$

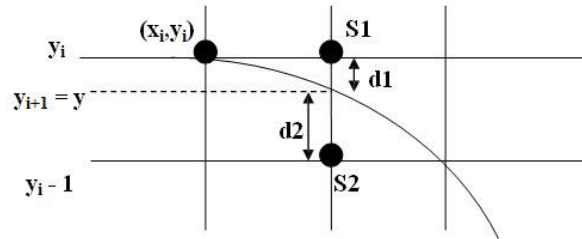


Hình 2.11. Sơ đồ khối thuật toán MidPoint vẽ đường tròn

Minh họa thuật toán MidPoint trong phần phụ lục luận văn.

Vẽ đường tròn bằng thuật toán Bresenham

Tương tự thuật toán vẽ đường thẳng Bresenham, các vị trí ứng với các tọa độ nguyên nằm trên đường tròn có thể tính được bằng cách xác định một trong hai pixel gần nhất với đường tròn thực hơn trong mỗi bước.



Hình 2.12. Đường tròn với khoảng cách d_1 và d_2 .

Ta có:

$$\begin{aligned} d_1 &= (y_i)^2 - y_2^2 = (y_i)^2 - (R_2 - (x_i + 1)^2) \\ d_2 &= y_2^2 - (y_i - 1)^2 = (R_2 - (x_i + 1)^2) - (y_i - 1)^2 \\ P_i &= d_1 - d_2 \end{aligned}$$

Tính $P_{i+1} - P_i$

$$\Rightarrow P_{i+1} = P_i + 4x_i + 6 + 2((y_i+1)^2 - (y_i)^2) - 2(y_{i+1} - y_i)$$

- Nếu $P_i < 0$: chọn $y_{i+1} = y_i$. Khi đó $P_{i+1} = P_i + 4x_i + 6$

- Nếu $P_i \geq 0$: chọn $y_{i+1} = y_i - 1$. Khi đó $P_{i+1} = P_i + 4(x_i - y_i) + 10$.

- P_0 ứng với điểm ban đầu $(x_0, y_0) = (0, R)$ là: $P_0 = 3 - 2R$.

Minh họa thuật toán vẽ đường tròn bằng Bresenham trình bày trong phần phụ lục của luận văn.

2.1.2.7. Thuật toán vẽ Ellipse

* Đầu vào :

- Cho tọa độ điểm $Q (X_q, Y_q)$; a ; b

* Đầu ra :

- Ellipse .

* Thuật toán

Tương tự thuật toán vẽ đường tròn, sử dụng thuật toán Bresenham để vẽ, ta chỉ cần vẽ 1/4 ellipse, sau đó lấy đối xứng qua các trục tọa độ sẽ vẽ được toàn bộ ellipse.

B1- Xét ellipse có tâm O , các bán kính là a và b , phương trình là: $(x_2/a_2) + (y_2/b_2) = 1$

B2- Chọn tọa độ pixel đầu tiên cần hiển thị là $(x_i, y_i) = (0, b)$. Cần xác định pixel tiếp theo là (x_{i+1}, y_{i+1}) . Ta có:

$$\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} = \begin{cases} y_i - 1 \\ y_i \end{cases} \end{cases}$$

$$\begin{aligned} d_1 &= (y_i)^2 - y_2 \\ d_2 &= y_2 - (y_i - 1)^2 \\ P_i &= d_1 - d_2 \end{aligned}$$

B3- Tính $P_{i+1} - P_i$

$$\Rightarrow P_{i+1} = P_i + 2((y_{i+1})^2 - (y_i)^2) - 2(y_{i+1} - y_i) + (2b_2/a_2)(2x_i + 3)$$

- Nếu $P_i < 0$: chọn $y_{i+1} = y_i$. Khi đó $P_{i+1} = P_i + (2b_2/a_2)(2x_i + 3)$
- Nếu $P_i \geq 0$: chọn $y_{i+1} = y_i - 1$. Khi đó $P_{i+1} = P_i + (2b_2/a_2)(2x_i + 3) + 4(1 - y_i)$
- P_i ứng với điểm ban đầu $(x_0, y_0) = (0, b)$ là: $P_0 = (2b_2/a_2) - 2b + 1$

Minh họa thuật toán vẽ Ellipse trình bày trong phần phụ lục của luận văn.

Thuật toán vẽ đường conics và một số đường cong khác

* Đầu vào :

- Đường conics có dạng: $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$
- Hằng số A, B, C, D, E, F

* Đầu ra :

- Đường conics .

* Thuật toán

Phương trình tổng quát của các đường conics có dạng: $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$

Giá trị của các hằng số A, B, C, D, E, F sẽ quyết định dạng của đường conics, cụ thể là nếu:

- $B^2 - 4AC < 0$: dạng đường tròn (nếu $A=C$ và $B=0$) hay ellipse.
- $B^2 - 4AC = 0$: dạng parabol.
- $B^2 - 4AC > 0$: dạng hyperbol.

Áp dụng ý tưởng của thuật toán Midpoint để vẽ các đường conics và một số đường cong khác theo các bước theo các bước tuần tự sau:

Bước 1: Dựa vào dáng điệu và phương trình đường cong, để xem thử có thể rút gọn phần đường cong cần vẽ hay không.

Bước 2: Tính đạo hàm, từ đó phân thành các vùng vẽ.

+ Nếu $0 \leq f'(x) \leq 1$: $x_{i+1} = x_i + 1$; $y_{i+1} = y_i$ (hoặc $= y_{i+1}$)

+ Nếu $-1 \leq f'(x) \leq 0$: $x_{i+1} = x_i + 1$; $y_{i+1} = y_i$ (hoặc $= y_{i-1}$)

+ Nếu $f'(x) > 1$: $y_{i+1} = y_i + 1$; $x_{i+1} = x_i$ (hoặc $= x_{i+1}$)

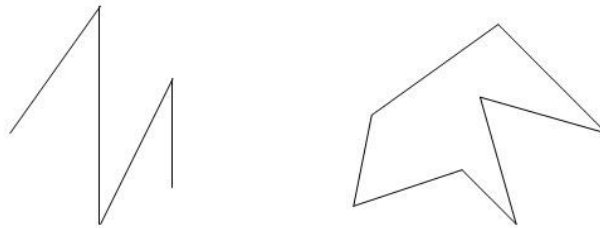
+ Nếu $f'(x) < -1$: $y_{i+1} = y_i + 1$; $x_{i+1} = x_i$ (hoặc $= x_{i+1}$)

Bước 3: Tính P_i cho từng trường hợp để quyết định $f'(x)$ dựa trên dấu của P_i . P_i thường là hàm được xây dựng từ phương trình đường cong. Cho $P_i=0$ nếu (x_i, y_i) thuộc về đường cong. Việc chọn P_i cần chú ý sao cho các thao tác tính P_i sau này hạn chế phép toán trên số thực.

Bước 4: Tìm mối liên quan của P_{i+1} và P_i bằng cách xét hiệu $P_{i+1} - P_i$

Bước 5: Tính P_0 và hoàn chỉnh thuật toán.

2.1.2.8. Vẽ đa giác



Hình 2.13. Hai dạng của đường gấp khúc.

* Đầu vào :

Tọa độ các đỉnh A, B, C, D, E, F ..

* Đầu ra :

Đa giác

* Thuật toán

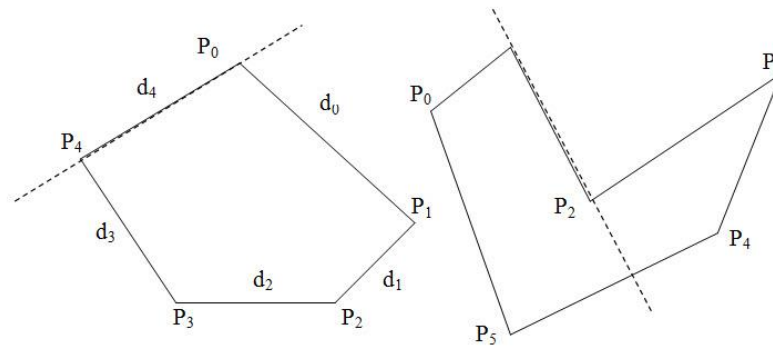
Đa giác là một đường gấp khúc kín có đỉnh đầu và đỉnh cuối trùng nhau.

Xây dựng cấu trúc dữ liệu để vẽ đa giác

```
Type
d_dinh = record x,y: longint;
var
end;
dinh = array[0..10] of d_dinh;
d: dinh;
```

B1- Với cách xây dựng cấu trúc dữ liệu như thế này thì chúng ta chỉ cần nhập vào tọa độ các đỉnh và sau đó gọi thủ tục vẽ đường thẳng lần lượt qua 2 đỉnh như (0, 1), (1,2),..., (n-1, n), trong đó đỉnh n trùng với đỉnh 0 thì ta sẽ vẽ được toàn bộ đa giác.

B2- Đa giác được gọi là lồi nếu bất kỳ đường thẳng nào đi qua một cạnh của đa giác thì toàn bộ đa giác nằm về một phía của đường thẳng đó. Ngược lại, nếu tồn tại ít nhất một cạnh của đa giác chia đa giác làm 2 phần thì gọi là đa giác lõm.



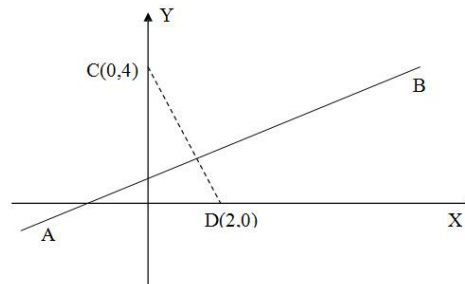
Hình 2.14. Đa giác lồi và đa giác lõm

Thuật toán kiểm tra một đa giác là lồi hay lõm

Thuật toán 1: Lần lượt thiết lập phương trình đường thẳng đi qua các cạnh của đa giác. Ứng với từng phương trình đường thẳng, xét xem các đỉnh còn lại có nằm về một phía đối với đường thẳng đó hay không? Nếu đúng thì kết luận đa giác lồi, ngược lại là đa giác lõm.

Nhận xét: Phương trình đường thẳng $y = ax + b$ chia mặt phẳng ra làm 2 phần. Các điểm nằm $C(x_c, y_c)$ trên đường thẳng sẽ có $y_c > ax_c + b$ và các điểm $D(x_d, y_d)$ nằm phía dưới đường thẳng sẽ có $y_d < ax_d + b$.

Cho đường thẳng AB có phương trình $y = \text{độ}$ là $C(0,4)$, $D(2,0)$.



Hình 2.15. Đường thẳng AB và 2 điểm C, D.

Vậy hai điểm C, D nằm về hai phía đối với đường thẳng AB.

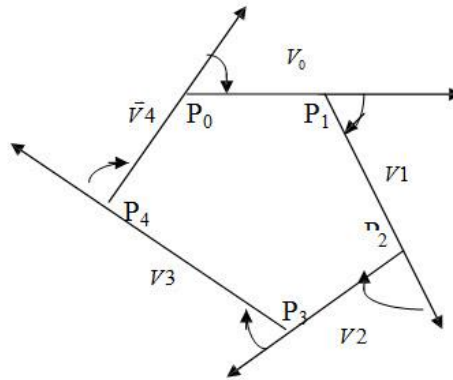
Thuật toán 2

Nhận xét: Trong mặt phẳng Oxy, cho 2 véc tơ a_n và b_n , Tích vô hướng của 2 véc tơ là:

$$T(\vec{a}, \vec{b}) = \begin{vmatrix} a_x & a_y \\ b_x & b_y \end{vmatrix} = a_x * b_y - a_y * b_x$$

- a_n rẽ trái sang b_n nếu $T \geq 0$
- a_n rẽ phải sang b_n nếu $T < 0$

Một đa giác là lồi khi đi dọc theo biên của nó thì chỉ đi theo một hướng mà thôi. Nghĩa là chỉ rẽ phải hay rẽ trái. Ngược lại là đa giác lõm.



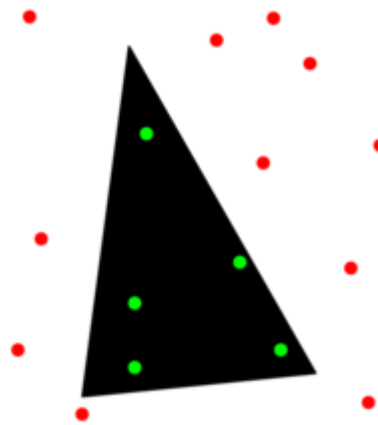
Hình 2.16. Đa giác lồi có 5 đỉnh.

Xét đa giác gồm các đỉnh P_0, P_1, \dots, P_n , ($P_0 = P_n$), $n \geq 3$. Tính $V_i = P_{i+1} - P_i$, $\forall i = 0, 1, \dots, n-1$. Tính $T_i = T(V_i, V_{i+1})$. Nếu với mọi T_i đều cùng dấu thì kết luận đa giác lồi. Ngược lại, là đa giác lõm.

2.2. Tìm kiếm vùng

2.2.1. Tìm kiếm vùng đơn hình

Dạng tổng quát nhất của bài toán tìm kiếm vùng là như sau: xử lý và lưu trữ một tập hợp S các đối tượng, sao cho có thể xác định xem một vùng cho trước chứa những đối tượng nào trong S . Chẳng hạn S có thể là một tập hợp các điểm tương ứng với tọa độ của các thành phố, và ta muốn tìm xem có những thành phố nào trong khoảng kinh độ và vĩ độ cho trước.



Hình 2.17. Vùng

Bài toán tìm kiếm vùng và các cấu trúc dữ liệu cho nó là những vấn đề cơ bản của hình học tính toán. Bài toán tìm kiếm vùng không chỉ có ứng dụng trong xử lý dữ

liệu hình học (như trong hệ thống thông tin địa lý (GIS) hay CAD, mà còn trong cơ sở dữ liệu nói chung.

2.2.2. Các biến thể

Có nhiều biến thể khác nhau của bài toán này, và các biến thể khác nhau đòi hỏi các cấu trúc dữ liệu khác nhau. Để có được lời giải hiệu quả, cần xác định các yếu tố sau của bài toán:

- Loại đối tượng: thuật toán tùy thuộc vào S bao gồm các điểm, đường thẳng, đoạn thẳng, hình chữ nhật, đa giác,... Đối tượng đơn giản nhất và cũng được nghiên cứu nhiều nhất là điểm.
- Loại vùng: Các vùng cần tìm phải thuộc một thể loại nhất định. Một vài thể loại được nghiên cứu cùng với tên của bài toán tương ứng bao gồm hình chữ nhật song song trục tọa độ (tìm kiếm vùng trục giao), đơn hình (tìm kiếm vùng đơn hình), nửa không gian (tìm kiếm vùng nửa không gian), và hình cầu (tìm kiếm vùng hình cầu).
- Loại câu hỏi: có một số loại câu hỏi khác nhau thường được nghiên cứu: liệt kê tất cả các đối tượng trong vùng, đếm số đối tượng trong vùng, kiểm tra xem có tồn tại đối tượng nào trong vùng hay không.
- Tìm kiếm vùng động và tìm kiếm vùng tĩnh: trong trường hợp tĩnh, tập hợp S được biết ngay từ đầu và không thay đổi. Trong trường hợp động, tập hợp S có thể được chèn thêm hoặc xóa bớt giữa các câu hỏi.
- Tìm kiếm vùng offline: cả tập hợp đối tượng và tập hợp các câu hỏi được biết đến ngay từ đầu.

2.3. Thuật toán Ray Casting

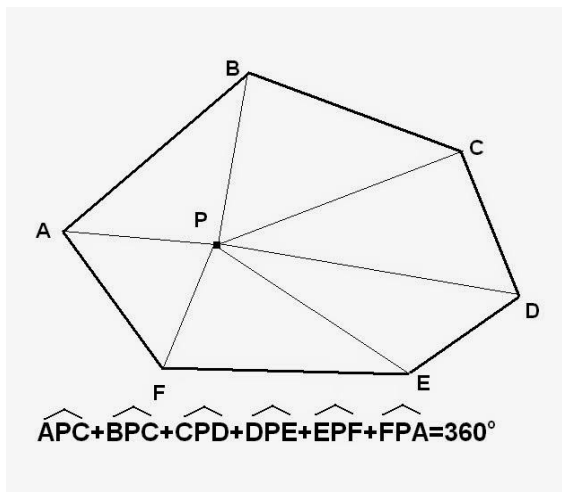
2.3.1. Kiểm tra một điểm trong một đa giác trên mặt phẳng tọa độ

Hẳn không chỉ trong lý thuyết hay trong giải bài tập Toán hình học mà chúng ta gặp phải một câu hỏi: "làm sao để biết 1 điểm với tọa độ (x,y) có nằm trong một hình đa giác (lồi) hay không?", hơn nữa chúng ta còn thường xuyên phải giải bài toán này trên máy tính, hoặc trong những lĩnh vực khác. Chẳng hạn, bạn cần viết một ứng dụng cho phép vẽ các hình bất kì, và sau khi vẽ có thể tự do chọn lựa lại những hình đó và di chuyển chúng trên màn hình bằng chuột. Chắc chắn rằng bạn sẽ gặp trở ngại khi cố tìm cách làm sao để máy tính hiểu người dùng đang kéo chuột tới hình nào. Nên nhớ rằng màn hình hiển thị của máy tính cũng là một hệ trục tọa độ 2 chiều, và mọi hình phẳng đều có thể xấp xỉ bằng những hình đa giác lồi. Vậy, chúng ta phải giải quyết bài toán đơn giản với trước khi nghĩ tới những hình phẳng có hình thù kì quái hơn.

Có nhiều cách để giải bài toán xác định một điểm có nằm trong một đa giác lồi. Về mặt lý thuyết, ta có một số cách sau đây:

2.3.1.1. Kiểm tra góc

Nối lần lượt các đường thẳng từ điểm đó đến các đỉnh của đa giác. Thực hiện cộng tất cả các số đo góc của góc tạo bởi 2 đỉnh liền kề với điểm đó theo thứ tự các đỉnh. Nếu tổng này bằng 360 độ thì điểm đó nằm trong đa giác.



Hình 2.18. Điểm P nằm trong đa giác ABCDEF, thỏa điều kiện trên.

2.3.1.2. Kiểm tra diện tích

Cũng với ý tưởng của cách trên, nhưng thay vì tính góc, ta sẽ tính diện tích của từng tam giác nhỏ tạo bởi điểm đang xét và 2 đỉnh kề nhau. Dem so sánh tổng diện tích của những hình tam giác nhỏ đó với diện tích của đa giác lớn.

2.3.1.3. Kiểm tra cùng phía của điểm và các cạnh đa giác

Lấy ra một cạnh và kéo dài thành 1 tia rồi xét điểm đang xét đó có nằm cùng phía với cái đỉnh còn lại không. Nếu với tất cả các cạnh của đa giác, điểm đó đều nằm cùng phía thì thuộc đa giác.

2.3.1.4. Thuật toán Ray Casting

* Đầu vào :

Điểm M, Đa giác H

* Đầu ra :

Xác định vị trí M so với đa giác

* Thuật toán

B1- Cho 3 điểm P,Q,R. kiểm tra Q nằm trong đoạn QR;

B2- Kiểm tra hướng của bộ ba thứ tự P, Q, R;

- Kết quả:(0): thẳng hàng; (1) theo chiều kim đồng hồ; (2) ngược lại
- Tìm 4 hướng cần thiết đối với trường hợp tổng quát và đặc biệt.
- Điểm M thuộc vào đa giác, đa giác H có n đỉnh

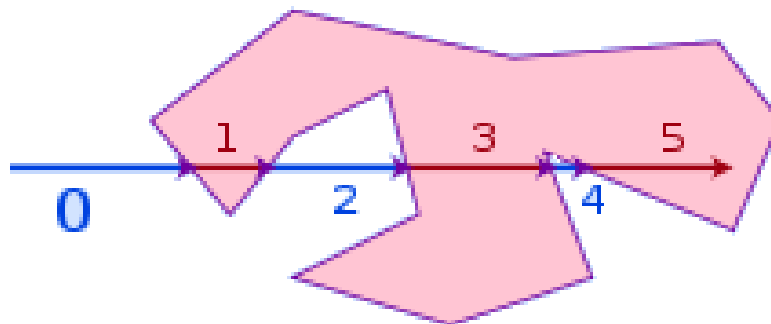
B3- Để kiểm tra đoạn thẳng Từ P đến đầu kia cắt với đoạn thẳng từ M giao điểm của đoạn trên với các cạnh của đa giác.

- Nếu điểm P thẳng hàng với đoạn tiếp theo thì cần kiểm tra nó nằm trong đoạn nào?

Theo [7], xét một tia (ra) bắt đầu từ điểm đó giao với các cạnh của đa giác bao nhiêu lần Với cách này, ta có thể áp dụng với một đa giác không lồi, hay một miền không phải hình sao. Nếu điểm không nằm trên biên của đa giác thì ta sẽ có 2 trường hợp:

Điểm nằm ngoài đa giác nếu tổng số giao điểm là số chẵn

Điểm nằm trong đa giác nếu tổng số giao điểm là số lẻ.



Hình 2.19. ý tưởng cho thuật toán Ray Casting

Ta biết rằng con trỏ hiển thị trên màn hình được điều khiển bằng thiết bị được gọi là chuột, và tại mỗi thời điểm con trỏ này mang một tọa độ ứng với hệ trục tọa độ của màn hình máy vi tính. Trong thời kỳ đầu của đồ họa máy tính, vấn đề kiểm tra một điểm trong một hình đa giác thường được tiếp cận theo hai cách: quét tia, tức cách 4 và phép cộng góc, tức cách 1.

2.3. Kết luận chương

Chương 2 trình bày những thuật toán liên quan đến ứng dụng trong chương sau.

Bắt đầu từ biểu diễn điểm, đường... trong tọa độ Decac, chương trên đề cập các thuật toán liên quan đến xác định giao của đoạn thẳng, đường thẳng, xác định vị trí tương đối giữa các yếu tố hình học.

Ngoài ra chương 2 đã trình bày thuật toán về vẽ các yếu tố hình học, như vẽ đường thẳng, đa giác, đường cong, hình tròn, hình ellipse...

Chương 3 sẽ sử dụng các thuật toán này trong bài toán xác định điểm so với vùng đa giác, trên hệ thống bản đồ.

Chương 3

Khái niệm bản đồ

Chương này sẽ trình bày một số khái niệm hệ thống bản đồ, có khả năng ứng dụng trong thực tế.

Trước hết chương này đề cập hệ thống bản đồ số, rồi sử dụng thuật toán Ray Casting để xác định một điểm thuộc vào vùng giới hạn theo đa giác.

3.1. Bản đồ

3.1.1. Khái niệm bản đồ

Bản đồ là hình thu nhỏ tương đối chính xác về một khu vực hay cả Trái Đất. Bản vẽ đơn giản miêu tả một không gian, địa điểm và hiển thị những thông số liên quan trực tiếp đến vị trí ấy có liên quan đến khu vực xung quanh.

Theo các nhà bản đồ: Bản đồ là sự miêu tả khái quát, thu nhỏ bề mặt Trái Đất hoặc bề mặt thiên thể khác trên mặt phẳng trong một phép chiếu xác định, nội dung của bản đồ được biểu thị bằng hệ thống ký hiệu quy ước.

Bản đồ thường dùng nhất trong địa lý. Theo nghĩa này bản đồ thường có hai chiều mà vẫn biểu diễn một không gian có ba chiều đúng đắn. Môn bản đồ là khoa học và nghệ thuật vẽ bản đồ.

Bản đồ còn là một khái niệm được sử dụng trong sinh học để biểu thị một hệ thống nào đó, ví dụ như bản đồ gen.

Tỉ lệ

Tỉ lệ của một bản đồ địa lí là tỉ số giữa một khoảng cách đo trên bản đồ và khoảng cách ngoài thực địa.

- Chẳng hạn, nếu 1 cm trên bản đồ ứng với 1 km ngoài thực địa thì bản đồ đó có tỉ lệ 1:100000, vì $1 \text{ km} = 100000 \text{ cm}$.
- Kí hiệu của tỉ lệ có dạng 1:M, trong đó số M chỉ khoảng cách thực tế lớn gấp bao nhiêu lần khoảng cách tương ứng đo trên bản đồ.

Bản đồ có tỉ lệ lớn thì càng chi tiết và tương ứng với số M nhỏ. Bản đồ tỉ lệ nhỏ kém chi tiết hơn và có số M lớn.

3.1.2. Bản đồ địa chính

Bản đồ địa chính ra đời từ rất sớm (thời Napoleon), bắt đầu ở Pháp, rồi lan đến các nước Châu Âu, nước Mỹ, Canada,... để nhằm mục đích là kiểm kê đất đai và thu thuế.

3.1.2.1. Khái niệm bản đồ địa chính

Bản đồ địa chính (Cadastral Map) là bản đồ trên đó thể hiện các dạng đồ họa và ghi chú, phản ánh những thông tin về vị trí, ý nghĩa, trạng thái pháp lý của các thửa đất, phản ánh các đặc điểm khác thuộc địa chính quốc gia.

Bản đồ địa chính là bản đồ chuyên ngành đất đai trên đó thể hiện chính xác vị trí ranh giới, diện tích và một số thông tin địa chính của từng thửa đất, vùng đất. Bản đồ địa chính còn thể hiện các yếu tố địa lý khác liên quan đến đất đai được thành lập theo đơn vị hành chính cơ sở xã, phường, thị trấn và thống nhất trong phạm vi cả nước.

3.1.2.2. Mục đích của bản đồ địa chính

Bản đồ địa chính được thành lập với những 4 mục đích chính như sau:

1. Thống kê, kiểm kê diện tích đất đai từng khu vực và trong cả nước.
2. Xác lập quyền sở hữu hoặc quyền sử dụng đất trên từng lô đất cụ thể của nhà nước và mọi công dân.
3. Là công cụ giúp nhà nước thực thi các nhiệm vụ, công việc có liên quan đến đất đai: thu thuế, giải quyết tranh chấp, quy hoạch đất đai, đền bù,...
4. Cung cấp thông tin về đất đai và cơ sở pháp lý cho các hoạt động dân sự như: thừa kế, chuyển nhượng, cho, tặng, thế chấp, kinh doanh bất động sản...

3.1.2.3. Nội dung của bản đồ địa chính

Nội dung cơ sở địa lý

- Yếu tố cơ sở toán học: bao gồm khung bản đồ, lưới bản đồ, các điểm khống chế, tỷ lệ bản đồ, sơ đồ phân mảnh.
- Yếu tố thủy văn: biểu thị ranh giới, tên gọi, mối quan hệ tương hỗ của các yếu tố như sông ngòi, ao, hồ, kênh mương...
- Yếu tố dáng đất: là tập hợp những chỗ lồi lõm trên bề mặt Trái đất. Địa hình được biểu thị lên bản đồ địa chính bằng các điểm độ cao (đối với khu vực đồng bằng), bằng các điểm độ cao kết hợp đường bình độ (khu vực miền núi). Phải thể hiện được dáng đất chung của địa hình toàn khu vực và các nét đặc trưng

của nó bằng việc lựa chọn khoảng cao đều đường bình độ. Địa hình phải được thể hiện phù hợp với các yếu tố khác như thủy hệ, giao thông...

- Yếu tố kinh tế – văn hóa – xã hội: thể hiện những địa vật kinh tế, văn hóa, xã hội mang tính chất định hướng trong khu vực thành lập bản đồ như đình, chùa, trạm biến thế, ngã ba, ngã tư... Ngoài ra tất cả các điểm địa vật có ý nghĩa quan trọng với sự phát triển kinh tế, văn hóa, xã hội cũng phải được thể hiện đầy đủ như các bệnh viện, trường học... Tuy nhiên, số lượng các địa vật phụ thuộc vào tỷ lệ bản đồ và quy phạm quy định của bản đồ tỷ lệ tương ứng.
- Yếu tố giao thông: biểu thị tất cả các đường giao thông và các yếu tố có liên quan như đường bộ, đường thủy, đường sắt, đường hàng không (chỉ biểu thị tên gọi).
- Ranh giới, địa giới hành chính: biểu thị chính xác, đầy đủ ranh giới quốc gia, ranh giới tỉnh/thành phố, ranh giới quận/huyện, phường/xã, vùng khu vực nhỏ như cơ quan, trường học... Các mốc địa giới hành chính được xác định tọa độ và được thể hiện lên trên bản đồ. Đối với các đơn vị hành chính giáp biển, các đảo nếu trong hồ sơ địa giới hành chính không khép kín ranh giới hành chính thì trên bản đồ hành chính thể hiện đến ranh giới sử dụng đất tiếp giáp với phần biển.

Các yếu tố cơ bản

1. Ranh giới thửa đất: là yếu tố chính và rất quan trọng của nội dung bản đồ địa chính, được hiển thị bằng đường viền khép kín, nét liền theo hệ thống kí hiệu của bản đồ.
2. Số hiệu thửa và diện tích đất: Số hiệu thửa được ghi cho mỗi thửa là duy nhất không trùng lặp trong phạm vi một tờ bản đồ địa chính và tương ứng với một chủ hoặc một đồng chủ sử dụng đã được xác minh về mặt pháp lý. Diện tích đất được xác định chính xác đến 0.1 m².
3. Loại đất: được chia làm 3 nhóm chính đất nông nghiệp, đất phi nông nghiệp và nhóm đất chưa sử dụng. Trên bản đồ, loại đất được thể hiện bằng kí hiệu chữ theo quy phạm.
4. Các yếu tố tự nhiên, nhân tạo có trên thửa đất: ở khu vực đô thị và các khu vực của tổ chức nhà nước giao đất, cho thuê đất chỉ thể hiện các công trình chính không thể hiện các công trình tạm thời, ở khu vực nông thôn không thể hiện các công trình xây dựng.

3.1.3. Bàn đồ số

3.1.3.1. Khái niệm

Số hóa là hình thức chuyển đổi các dữ liệu truyền thống bên ngoài thành dạng dữ liệu số mà máy tính có thể hiểu được.

Thông thường, các dữ liệu dạng chữ, hình ảnh, âm thanh... sử dụng trên máy tính và được máy tính nhận biết đúng định dạng, được gọi chung là dữ liệu số. Quá trình chuyển các dạng dữ liệu truyền thống như các bản viết tay, bản in trên giấy, hình ảnh... sang chuẩn dữ liệu trên máy tính và được máy tính nhận biết được gọi là số hoá dữ liệu.

Như vậy, số hoá dữ liệu là hình thức chuyển đổi các dữ liệu truyền thống bên ngoài thành dạng dữ liệu số mà máy tính có thể hiểu được.

3.1.3.2. Lí do phải số hóa dữ liệu

Việc lưu trữ thông tin trên giấy tờ sẽ rất tốn kém dữ liệu, hơn nữa việc bảo quản và phạm vi sử dụng bị hạn chế. Do vậy bắt buộc chúng ta phải nghĩ đến giải pháp số hóa dữ liệu. Việc số hóa dữ liệu sẽ giúp việc lưu trữ, truy xuất, chi sẻ, tìm kiếm thông tin một cách nhanh chóng và dễ dàng nhất.

3.1.3.3. Ưu điểm và hạn chế

Ưu điểm

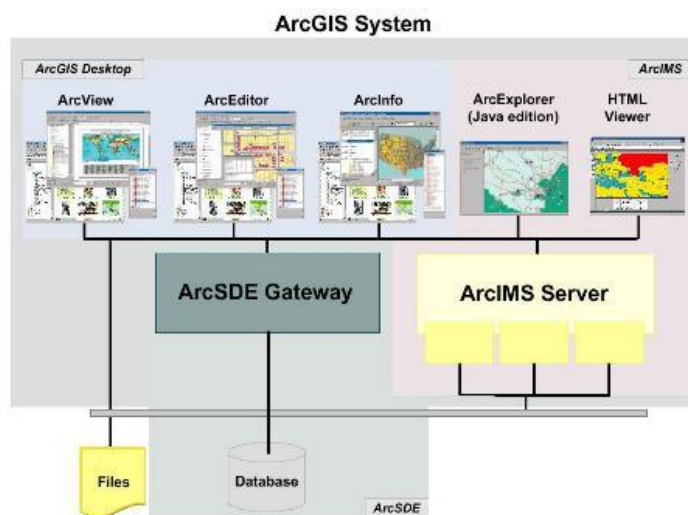
- Giúp việc lưu trữ, truy xuất, chia sẻ, tìm kiếm thông tin một cách dễ dàng
- Linh hoạt trong việc chuyển đổi sang các loại dữ liệu số khác nhau
- Giảm chi phí tối đa cho việc quản lý, không gian lưu trữ
- Có khả năng chỉnh sửa và tái sử dụng dữ liệu

Hạn chế

- Cần đầu tư ban đầu về công nghệ, cơ sở hạ tầng CNTT, máy móc hiện đại.
- Dữ liệu dễ bị sao chép và sửa đổi trái pháp luật.
- Việc triển khai sử dụng gặp nhiều khó khăn do phải thực hiện training đồng bộ và có hệ thống. Ngoài ra việc bảo mật dữ liệu cũng là một thách thức lớn.

3.1.4. ArcGIS, giải pháp toàn diện cho hệ thống thông tin địa lý

Theo [5], hãng ESRI, Mỹ là một trong những hãng tiên phong trong lĩnh vực hệ thống thông tin địa lý và có sản phẩm nổi tiếng trước đây là ArcInfo và hiện nay là hệ thống ArcGIS. ArcGIS là một hệ thống phần mềm cung cấp một giải pháp tổng thể về hệ thống thông tin địa lý, bao gồm nhiều modul khác nhau, đáp ứng nhu cầu cho mọi tổ chức, từ những người sử dụng đơn lẻ cho đến hệ thống có tính toàn cầu.



Hình 3.1. ArcGIS, sẽ dùng trong luận văn

3.1.4.1. Xây dựng và quản trị cơ sở dữ liệu (CSDL) thông tin lưu trữ

Giải pháp lựa chọn cho quản trị CSDL về thông tin lưu trữ là sử dụng phần mềm Arc Spatial Data Engine (ArcSDE) của hệ thống ArcGIS. ArcSDE cho phép lưu trữ và quản lý thông tin theo mô hình cơ sở dữ liệu không gian (GeoDatabase) đa người sử dụng trong một hệ quản trị cơ sở dữ liệu quan hệ. Đây là một ưu điểm vượt trội của công nghệ GIS của hãng ESRI so với các hãng khác. ArcSDE là hạt nhân cho một môi trường đa người sử dụng của hệ thống ArcGIS, cung cấp một giao diện mở với các hệ quản trị cơ sở dữ liệu quan hệ và cho phép ArcGIS quản lý dữ liệu địa lý trong các nền các hệ quản trị cơ sở dữ liệu quan hệ như ORACLE, Microsoft SQL Server, IBM DB2, Informix. ArcSDE được lựa chọn do các ưu điểm sau:

- Có khả năng mô hình hóa các thông tin cần quản lý theo mô hình hướng đối tượng (Object Oriented Model). Thông tin quản lý theo mô hình hướng đối tượng: đảm bảo khả năng mô hình hóa tốt hơn, đơn giản hơn với người sử dụng, đảm bảo tốt hơn khả năng cập nhật, tính đồng nhất của dữ liệu, toàn vẹn dữ liệu và quan hệ giữa các đối tượng.
- Lưu trữ và quản lý tích hợp các dạng dữ liệu khác nhau trong một cơ sở dữ liệu duy nhất: bản đồ, hồ sơ, ảnh, mô hình số độ cao, bản vẽ thiết kế...
- Mô hình hóa cấu trúc topology trong CSDL không gian và cung cấp các chức năng nhằm bảo toàn quan hệ topology.
- Hệ thống tập trung với cơ chế truy nhập dữ liệu rộng rãi, đa người sử dụng đồng thời.
- Thực sự hoạt động theo kiến trúc khách - chủ 3 lớp và trên Internet.
- Có cơ chế quản lý truy nhập đồng thời trong một môi trường đa người sử dụng.

- Tối ưu hóa cho lưu trữ và tra cứu thông tin không gian, địa lý.
- Chuẩn hóa về quản trị dữ liệu như: sao lưu, phục hồi, sao chép dữ liệu.
- Mềm dẻo linh hoạt với dung lượng dữ liệu gần như không có hạn chế về kích thước.
- Có cơ chế đảm bảo an toàn và bảo mật dữ liệu.
- Duy trì chế độ bảo trì, nâng cấp theo thời gian.

3.1.4.2. Giải pháp xử lý các bài toán về quản lý, cập nhật, phân tích bản đồ

Giải pháp cho giải các bài toán về thiết kế, cập nhật, bảo trì CSDL không gian và phân tích xử lý bản đồ là phần mềm ArcInfo của ArcGIS. ArcInfo bao gồm ba ứng dụng chính: ArcMap, ArcCatalog và ArcToolBox. ArcInfo cho phép thực hiện bất cứ bài toán nào của GIS từ đơn giản cho đến phức tạp, bao gồm hiển thị bản đồ, quản lý dữ liệu, phân tích địa lý, sửa chữa và xử lý dữ liệu với dữ liệu được lưu trữ CSDL không gian (GeoDatabase).

ArcMap là ứng dụng thực hiện tất cả các nhiệm vụ về bản đồ bao gồm trình bày, hiển thị bản đồ, phân tích bản đồ và sửa chữa dữ liệu.

ArcCatalog là ứng dụng cho phép người sử dụng tổ chức và quản lý tất cả các dạng dữ liệu địa lý trong CSDL (CSDL không gian hoặc dạng file shape, coverage). ArcCatalog cung cấp các công cụ để hiển thị, tra cứu, tìm kiếm thông tin, ghi nhận và hiển thị thông tin metadata, định nghĩa lược đồ cấu trúc của các lớp thông tin địa lý.

ArcToolBox là ứng dụng cung cấp các công cụ Gis dùng cho phân tích và xử lý dữ liệu bản đồ như:

- Định nghĩa và chuyển hệ tọa độ.
- Phân tích, xử lý bản đồ: chồng xếp, thực hiện các phép toán đại số về bản đồ.
- Chuyển đổi khuôn dạng dữ liệu.

3.1.4.3. Giải pháp tổng hợp hóa, trình bày hiển thị tích hợp dữ liệu

Công tác tổng hợp hóa, trình bày và hiển thị dữ liệu rất quan trọng trong công tác quản lý đất đai. Một trong những ứng dụng hiệu quả nhất của công nghệ GIS là khả năng hiển thị đồng thời và tích hợp các dạng dữ liệu khác nhau theo mô hình 3 chiều. Dự án sẽ lựa chọn các phần mềm sau trong hệ thống ArcGIS để đáp ứng yêu cầu này:

ArcGIS 3D Analyst là ứng dụng dùng cho mô hình hóa, phân tích và xử lý bề mặt địa hình. ArcGIS 3D Ancalyst cung cấp các công cụ tiên tiến của GIS để thực hiện các bài toán trên bề mặt địa hình như: nội suy bình độ, tính mặt cắt, tính khối lượng, lưu vực...

3.1.4.4. Giải pháp tra cứu và phân phối thông tin trực tuyến trên mạng diện rộng Intranet/Internet

Hệ thống phần mềm ArcGIS đã đưa ra giải pháp tra cứu và cung cấp thông tin trực tuyến trên mạng diện rộng với phần mềm ArcInternet Map Server (ArcIMS). ArcIMS là một ứng dụng bản đồ rất mạnh trên Internet. Nó cung cấp một nền cơ bản cho phép xây dựng và phân phối các dịch vụ và dữ liệu GIS. ArcIMS bao gồm cả hai công nghệ chủ (server) và khách hàng (client). ArcIMS cung cấp giải pháp hiện đại cho tra cứu và phân phối dữ liệu bản đồ và thuộc tính trên mạng diện rộng (Intranet, Internet) theo giao diện Web. ArcIMS là một ứng dụng trên nền Java bao gồm các dịch vụ về bản đồ và các công cụ thiết kế bản đồ để cung cấp cho các dịch vụ về bản đồ trên Internet. ArcIMS cung cấp các mức độ bảo mật và an toàn thông tin khác nhau tùy thuộc vào yêu cầu của người sử dụng. Giải pháp của ArcIMS tuân theo các chuẩn, kiến trúc hiện đại, cung cấp cho người dùng GIS các khả năng xây dựng và quản trị các dịch vụ IMS.

3.1.4.5. Giải pháp xây dựng các phần mềm ứng dụng trên công nghệ gốc

Đây là một trong tiêu chí rất quan trọng để lựa chọn công nghệ gốc, thể hiện tính mở của hệ thống. Nó cho phép người sử dụng tạo ra những ứng dụng phù hợp với yêu cầu và đặc thù công việc.

- Tùy biến ứng dụng (customize): ArcGIS cho phép người sử dụng tùy biến trên công nghệ ArcObject. ArcObject là một tập hợp các thành phần phần mềm với các chức năng GIS và các giao diện có thể lập trình được. Công nghệ ArcObjects dựa trên giao thức COM. Các thành phần của ArcObjects làm cho ArcView, ArcEditor, ArcInfo và mô hình cơ sở dữ liệu không gian có tính mở với người sử dụng và nhà phát triển: thêm các công cụ mới vào phần mềm, định nghĩa các đối tượng của người sử dụng khi xây dựng CSDL không gian.
- Xây dựng các ứng dụng độc lập (User application): Công nghệ ArcInfo đưa giải pháp xây dựng các ứng dụng độc lập bằng bộ thư viện MapObjects. MapObjects là một thư viện các đối tượng thành phần của bản đồ và GIS cho phép phát triển xây dựng các ứng dụng độc lập có tương tác bản đồ và các chức năng của một hệ thống GIS. MapObjects là một điều khiển ActiveX với hơn 45 các đối tượng lập trình. MapObjects có khả năng truy nhập dữ liệu với các cơ sở dữ liệu cá nhân trên máy riêng lẻ hoặc với cơ sở dữ liệu không gian chạy trên môi trường đa người sử dụng.

MapObject là một môi trường lập trình tối ưu cho xây dựng các ứng dụng về GIS có tính đặc thù trong môi trường Windows. Giải pháp xây dựng các ứng dụng độc lập sử dụng để xây dựng phần mềm ứng dụng trên cơ sở dữ liệu tích hợp với GIS.

3.1.5. Quy trình lập bản đồ

3.1.5.1. Giới thiệu

Theo [1, 2] cùng với sự phát triển kỳ diệu của công nghệ thông tin trong những năm cuối thế kỷ XX, nhu cầu về số hoá và lượng hóa thông tin trên bản đồ ngày càng cao, đặc biệt là những bản đồ chuyên đề đã cung cấp những thông tin hữu ích để khai thác và quản lý tài nguyên. Ngoài ra trong xu thế hội nhập với khu vực và thế giới đòi hỏi những thông tin bản đồ phải phục vụ được nhiều ngành, nhiều lĩnh vực và nhiều đối tượng khác nhau, có khả năng trao đổi dữ liệu giữa các ngành với nhau. Những yêu cầu trên không thể thực hiện được đối với bản đồ giấy. Sự mô tả định lượng bị ngăn trở lớn do khối lượng số liệu và những quan trắc định lượng quá lớn. Ngoài ra hiện nay còn thiếu các công cụ quan trọng để mô tả sự biến thiên không gian mang tính chất định lượng. Vì vậy, việc thành lập bản đồ số, một trong những bước đi ban đầu trong việc xây dựng cơ sở dữ liệu địa chính quốc gia là rất cần thiết. Bản đồ số có thể được thành lập từ nhiều nguồn khác nhau: từ ảnh quét máy quét, từ ảnh hàng không, ảnh vệ tinh, từ các số liệu đo mặt đất. Sau đây là quá trình thành lập bản đồ số từ bản đồ giấy thông qua ảnh quét.

Bản đồ là một chỉnh thể bao gồm nhiều lớp thông tin chồng xếp lên nhau để mô tả thế giới thực. Thông tin trên bản đồ được phân ra thành 4 loại cơ bản sau:

- 1 *Đối tượng dạng điểm* (point): thể hiện các đối tượng chiếm diện tích nhỏ nhưng là thông tin rất quan trọng không thể thiếu như; trụ sở cơ quan, các công trình xây dựng, cầu cống...
- 2 *Đối tượng dạng đường* (line): thể hiện các đối tượng không khép kín hình học, chúng có thể là các đường thẳng, các đường gấp khúc và các cung, ví dụ như đường giao thông, sông, suối...
- 3 *Đối tượng dạng vùng* (region): thể hiện các đối tượng khép kín hình học bao phủ một vùng diện tích nhất định, chúng có thể là các polygon, ellipse và hình chữ nhật, ví dụ lãnh thổ địa giới 1 xã, hồ nước, khu rừng...
- 4 *Đối tượng dạng chữ* (text): thể hiện các đối tượng không phải là địa lý của bản đồ như nhãn, tiêu đề, ghi chú...

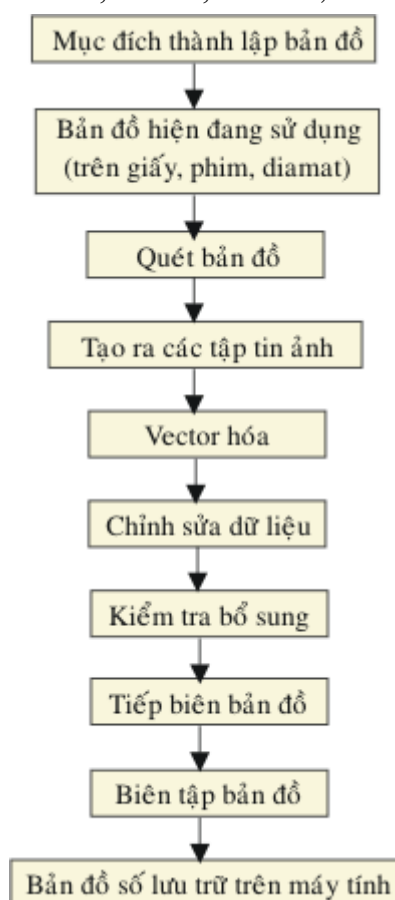
3.1.5.2. Mục đích

Bản đồ đang sử dụng hiện nay phần lớn là bản đồ giấy bao gồm rất nhiều loại khác nhau như: bản đồ địa hình, bản đồ địa chính, bản đồ hiện trạng sử dụng đất, bản đồ đất, và các bản đồ chuyên đề khác... Do đó tùy thuộc vào mục đích cụ thể mà thành lập bản đồ thích hợp. Tuy nhiên hiện nay Tổng cục địa chính quy định thống nhất dùng bản đồ địa hình và bản đồ địa chính làm bản đồ nền cơ sở trong toàn quốc. Do đó tất cả các bản đồ dù là thành lập với mục đích nào cũng đều được xây dựng trên nền bản đồ nền cơ sở trên.

Các đối tượng bản đồ khi tồn tại dưới dạng số được thể hiện và lưu trữ trên những lớp thông tin khác nhau. Vì vậy, trước khi số hóa thành lập bản đồ số, các đối tượng cần thể hiện trên bản đồ phải được xác định trước cần phải lưu trữ trên lớp thông tin nào.

3.1.5.3. Quét bản đồ

Quét bản đồ là quá trình chuyển các bản đồ được lưu trữ trên giấy, phim, diamat, thành các tập tin dữ liệu dưới dạng ảnh (raster file), sau đó tùy thuộc vào phần mềm xử lý ảnh và phần mềm quản lý bản đồ hiện có mà chuyển các raster file sang các định dạng khác như: *.TIFF, *.RLE, *.EPS, *.BMP,...



Hình 3.2. Quy trình lập bản đồ

Hiện nay trên thị trường có nhiều loại máy quét khác nhau về khổ giấy và về nhãn hiệu. Về khổ giấy thông dụng nhất là khổ A4, A3. Về nhãn hiệu phổ biến nhất là hiệu EPSON và một số nhãn hiệu của tập đoàn Inter-graph.

Cách thức hoạt động của máy quét: máy quét ghi nhận các ảnh bằng cách chiếu sáng vào tài liệu cần scan (bản đồ, văn bản), sau đó ánh sáng đi ngược trở lại và được tiếp nhận bởi một dãy các tế bào cảm quang gọi là thiết bị nạp đôi. Bởi vì các vùng tối trên giấy phản chiếu ít ánh sáng hơn và các vùng sáng của giấy phản chiếu nhiều hơn nên thiết bị nạp đôi có khả năng phát hiện ánh sáng phản chiếu ánh sáng từ mỗi vùng ảnh. Sau đó thiết bị nạp đôi sẽ chuyển các sóng ánh sáng được phản chiếu thành các thông tin dạng số, những thông tin này được biểu hiện bởi sự kết hợp của 2 số 0 và 1 (gọi là bit dữ liệu). Cuối cùng phần mềm quét sẽ đọc các dữ liệu mà máy nhận được và tái tạo nó thành một raster file lưu trữ trong máy tính.

Đây là giai đoạn rất quan trọng trong việc thành lập bản đồ số từ bản đồ giấy vì nó ảnh hưởng trực tiếp đến chất lượng ảnh thông qua việc chọn độ phân giải khi quét. Tuy nhiên, việc chọn độ phân giải cao hay thấp còn tùy thuộc vào nhiều yếu tố bao gồm: chất lượng tài liệu gốc, mục đích sử dụng, dung lượng trống của đĩa cứng. Cái giá phải trả cho một tệp raster có chất lượng cao là kích cỡ tệp raster đó sẽ lớn gây ra nhiều khó khăn cho việc lưu trữ và chuyển đổi.

3.1.5.4. Các dạng tệp Raster

Tùy thuộc vào phần mềm xử lý ảnh và phần mềm quản lý bản đồ hiện có mà chuyển thành các định dạng raster files khác nhau. Tuy nhiên, mỗi định dạng khác nhau đều có những thuận lợi và rắc rối riêng của nó. Sau đây là một số định dạng file:

- 1 *.TIFF (Tagged Image File Format) là dạng phổ biến nhất có khả năng lưu trữ các ảnh quét bằng nhiều độ phân giải, dạng màu và kiểu nén khác nhau, đặc biệt là thích nghi với nhiều trình ứng dụng.
- 2 *.EPS (Encapsulated Postscript) thích hợp cho dùng các bản vẽ vector nhưng lại không dùng cho lineart.
- 3 *.GIF là dạng dùng để lưu trữ các ảnh gồm 256 màu hoặc 256 các bóng xám.
- 4 *.PSP là dạng ảnh nội của Adobe Photoshop.
- 5 *.JPEG là dạng lưu trữ màu sắc hoặc các files thang độ xám.
- 6 *.PCX được sử dụng trong nhiều chương trình vẽ khác nhau cũng rất thích hợp cho các ảnh quét và rất thích nghi với cách sử dụng PC (máy tính cá nhân).

3.1.5.5. Nấn bản đồ

Đây là bước quan trọng nhất trong quy trình thành lập bản đồ số vì nó ảnh hưởng đến toàn bộ độ chính xác của bản đồ sau khi được số hóa dựa trên nền ảnh. Nấn bản đồ là quá trình chuyển đổi ảnh đang ở tọa độ hàng - cột của các điểm ảnh (pixel) về tọa độ trắc độ thực (hệ tọa độ địa lý hoặc tọa độ phẳng). Việc xác định tọa độ các điểm trên ảnh phải thật chính xác và trùng với bản đồ giấy. Tọa độ một điểm được xác định trên ảnh và thực tế có sự sai lệch nhau, tùy thuộc vào tỷ lệ bản đồ và mục đích thành lập bản đồ mà sai số cho phép sẽ khác nhau. Các điểm định vị trên vừa định nghĩa vùng làm việc cho quá trình số hóa, vừa là cơ sở cho quá trình tiếp biên giữa các mảnh bản đồ.

3.1.5.6. Vecto hóa

Vector hóa là quá trình biến đổi dữ liệu raster thành dữ liệu vector, hay nói cách khác đây là quá trình vẽ lại bản đồ giấy trên máy tính hoặc bản số hóa nhằm tạo một bản vẽ dạng số của bản đồ đó. Hiện nay có rất nhiều phần mềm số hóa bao gồm Autocad, Mapinfo, Arcinfo, Microstation... Sau khi số hóa, tùy thuộc vào phần mềm số hóa mà dữ liệu vector sẽ được tổ chức trong các định dạng tệp khác nhau như với Mapinfo sẽ được lưu trữ vào files*.TAB, với Microstation sẽ được lưu trữ vào files*.DGN.

3.1.5.7. Sửa dữ liệu

Sau quá trình số hóa, dữ liệu được nhận chưa phải đã hoàn thiện và sử dụng được, các dữ liệu này được gọi là dữ liệu thô, cần phải qua một quá trình chỉnh sửa hợp lệ. Quá trình này bao gồm các công đoạn: lọc bỏ điểm dư thừa, làm trơn đường, loại bỏ các đối tượng trùng nhau, sửa các điểm cuối tự do và tạo các điểm giao.

Sau khi chỉnh sửa dữ liệu là quá trình kiểm tra tính đầy đủ của đối tượng và độ chính xác của dữ liệu sau khi số hóa. Quá trình này ảnh hưởng đến độ chính xác cũng như chất lượng của sản phẩm sau này. Kiểm tra độ chính xác của dữ liệu là kiểm tra mức độ sai số giữa dữ liệu raster và dữ liệu vector (là độ lệch giữa các đường vector và tâm đường raster), thông thường sai số này phải $< 0,1$ mm tính theo tỷ lệ bản đồ. Kiểm tra tính đầy đủ đối tượng nghĩa là kiểm tra và bổ sung đầy đủ các đối tượng cần thu nhận theo yêu cầu đề ra đối với từng loại bản đồ tài liệu. Khi thực hiện công tác này người kiểm tra phải nắm được toàn bộ các thông số đồ họa quy định cho từng đối tượng, sử dụng thành thạo các công cụ sửa chữa và số hóa đối tượng để khi gặp các lỗi phải tiến hành xử lý ngay.

3.1.5.8. Tiếp biên

Không giống như các bản đồ trên giấy, công tác tiếp biên với các mảnh lân cận phải thực hiện ngay sau khi thu nhận và chỉnh sửa dữ liệu, các đối tượng dạng vùng tô màu phải chưa được tạo, bởi vì sau khi đóng vùng và tô màu nền, các yếu tố dạng vùng rất khó tiếp biên với nhau.

Dựa vào mục đích - yêu cầu của bản đồ cần thành lập, một lần nữa các đối tượng trên bản đồ được kiểm tra, thay đổi ký hiệu thích hợp và bố trí vị trí các đối tượng nhằm đảm bảo tính tương quan về địa hình cũng như tính thẩm mỹ của bản đồ.

Hiện nay có rất nhiều phần mềm xử lý bản đồ, cách tổ chức và quản lý dữ liệu không gian và dữ liệu thuộc tính ở các phần mềm có khác nhau, nhưng quy trình biên tập chuyển từ bản đồ giấy thành bản đồ số nhìn chung là giống nhau. Với sự phát triển của công nghệ thông tin cùng với nhu cầu về tổ chức xây dựng cơ sở dữ liệu quốc gia, hiện nay việc thành lập bản đồ số thay thế bản đồ giấy là rất cần thiết và là nhiệm vụ cấp bách.

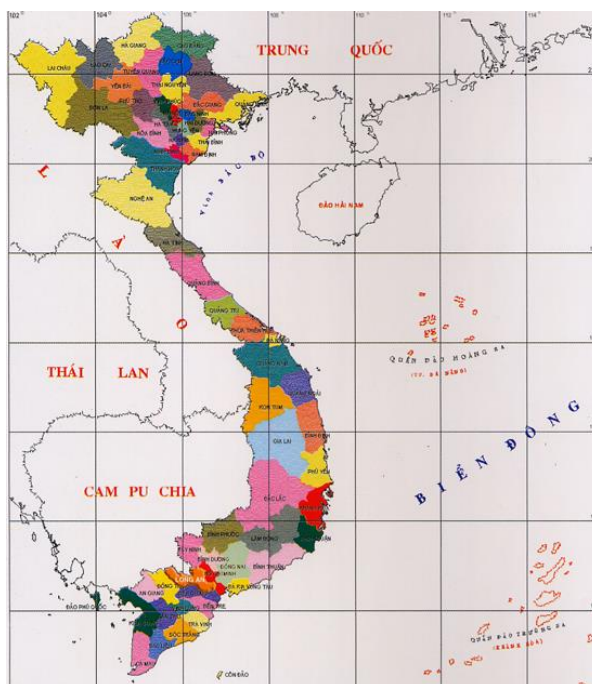
3.2. Ứng dụng trên bản đồ cần xác định điểm thuộc đa giác

3.2.1. Ứng dụng trên bản đồ địa chính

Bản đồ địa chính có ý nghĩa đối với mỗi người dân. Để đảm bảo sở hữu chính xác phần đất đai, công tác đo đạc và xác định các mốc thửa đất có ý nghĩa lớn.

Trên bản đồ địa chính, người ta có thể đặt các câu hỏi như “vị trí này có thuộc thửa đất của tôi không?”, hay “khu vực này có chạm vào nhà của ai không?”...

Những vấn đề như vậy được giải quyết nhờ xác định điểm có thuộc vùng đa giác hay không.



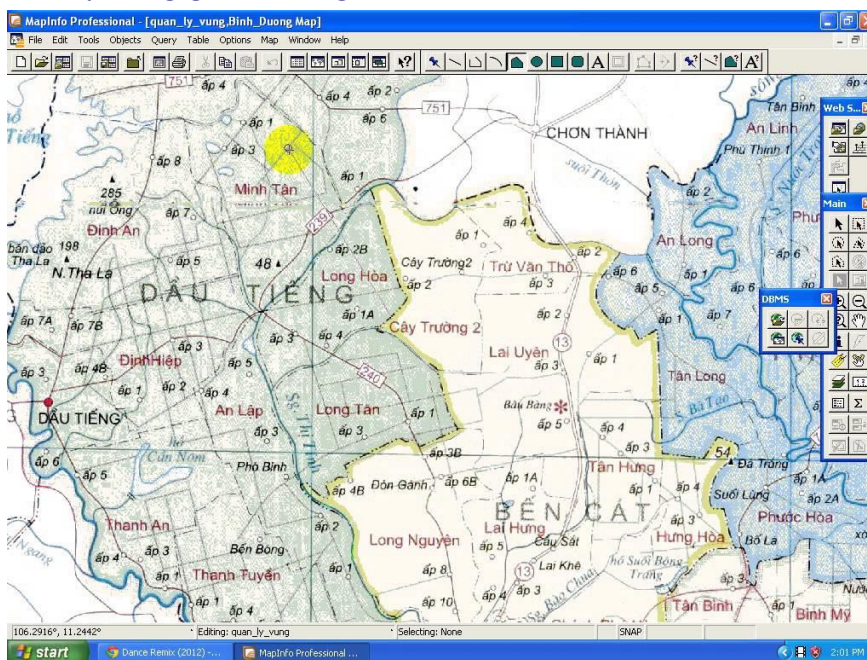
Hình 3.3. Bản đồ địa chính

Các bước thực hiện đối với vấn đề này :

1. Xác định tọa độ trên bản đồ địa chính;
2. Kiểm tra tọa độ X quan tâm có thuộc vào vùng đa giác;
3. Ra kết luận.

3.2.2. Ứng dụng trên bản đồ số

Bản đồ số là bản đồ tương tự được số hóa. Như vậy, các thông số trên bản đồ tương tự được thay bằng giá trị trong cơ sở dữ liệu về bản đồ.



Hình 3.4. Bản đồ số hóa trong MapInfo

Vấn đề vừa nêu trong phần trên được xác định lại theo các bước sau :

1. Xác định tọa độ của vùng đa giác. Thay vì đánh dấu trên bản đồ tương tự, cần xác định các tọa độ theo cơ sở dữ liệu bản đồ. Phần mềm quản trị bản đồ số hóa cho phép thực hiện việc này;
2. Điểm X được xác định theo tọa độ mà phần mềm quản trị bản đồ qui định;
3. Với các tọa độ, tức các bộ giá trị, ứng với hệ qui chiếu của phần mềm quản trị bản đồ số, cần áp dụng thuật toán hình học. Thuật toán này cho phép xác định X có thuộc vùng đa giác không.

Thực chất, ứng dụng trên bản đồ số khác với trên bản đồ địa chính, tương tự, ở chỗ các giá trị đại diện cho đối tượng địa lí được thay bằng các bộ dữ liệu trong hệ thống quản trị bản đồ.

3.2.3. Ứng dụng trên lãnh hải

Khu vực khai thác biển hay khu vực đặc quyền kinh tế Việt Nam có ý nghĩa trong vấn đề về hải phận Việt Nam. Ngay công ước quốc tế cũng chưa đủ để giải quyết tranh chấp lãnh hải.

Bài toán hình học áp dụng trong ứng dụng bản đồ có thể dùng trong bài toán trên lãnh hải. Cũng như trên đất, có thể sử dụng bản đồ tương tự hay bản đồ số.

Chẳng hạn đối với bản đồ số về vùng biển, bài toán xác định điểm X thuộc vào vùng đa giác, có một số trường hợp xảy ra :

- Xác định vị trí X của con tàu thuộc vào vùng lãnh hải Việt Nam;
- Xác định vị trí X của tàu, đối tượng gặp nạn, thuộc vào khu vực đang có cơ sở cứu hộ biển;
- Xác định vị trí X của đàn hải sản, so với vùng thả lưới của ngư dân;
- ...



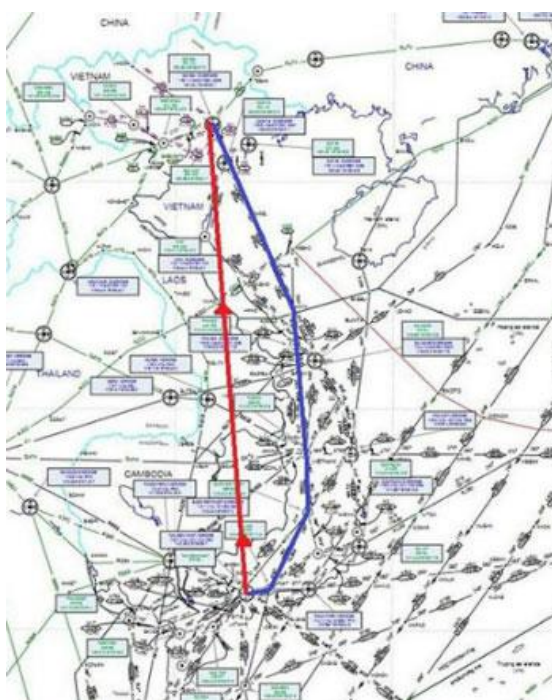
Hình 3.5. Ngư dân đánh cá trong khu vực qui định.

3.3.4. Ứng dụng trên không phận

Tai nạn trên không ngày nay không hiếm hoi. Một phần do ứng dụng không lưu đã đa dạng, phần khác do cần thiết có phương tiện giám sát, điều khiển các thiết bị bay.

Chẳng hạn trong tai nạn máy bay của hãng hàng không Malaysia 2015 cho thấy nhu cầu cần đến việc xác định điểm X thuộc vào vùng đa giác là xác định đối tượng X thuộc vào vùng không phận do Việt Nam quản lí.

Nhu cầu về bài toán hình học đối với không phận cũng như đối với hải phận.



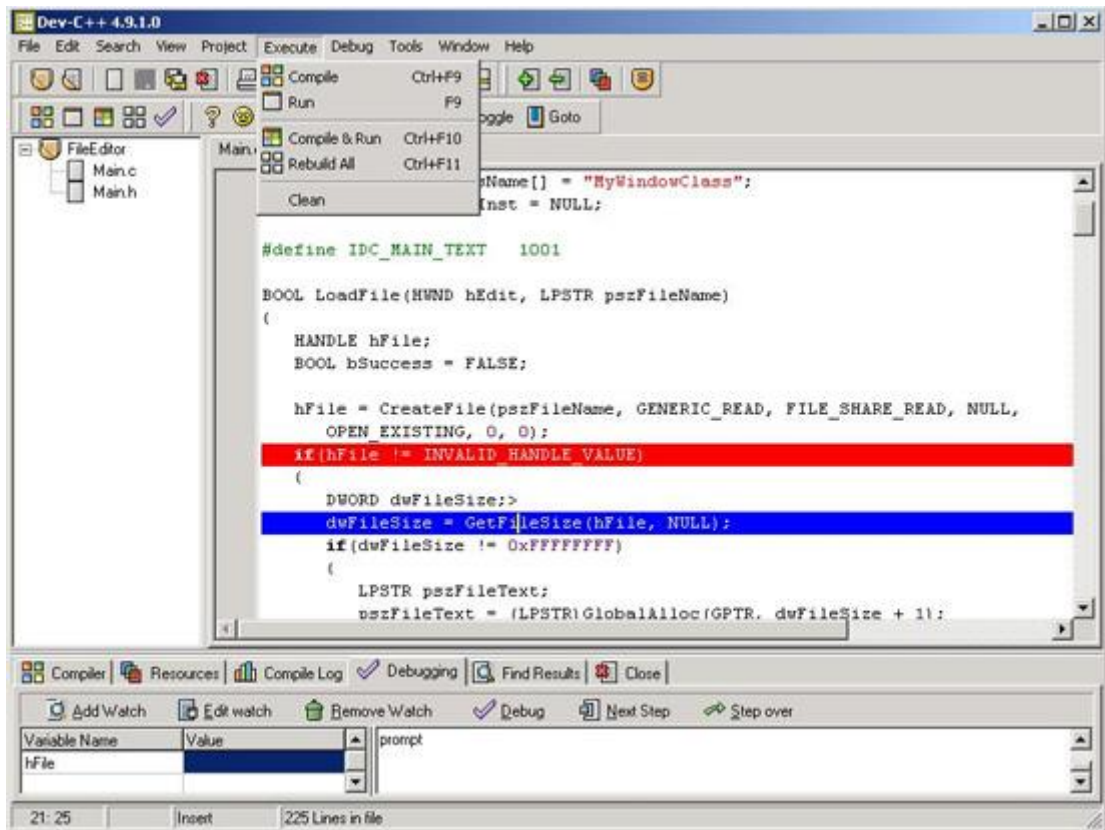
Hình 3.6. Lập đường bay trên không phận.

3.3. Kiểm tra một điểm thuộc vào đa giác nhờ thuật toán Ray Casting

3.3.1. Môi trường DEV C

Luận văn đã thử nghiệm trong môi trường DEV C++. Dev-C++ là một môi trường phát triển tích hợp tự do được phân phối dưới hình thức giấy phép Công cộng GNU hỗ trợ việc lập trình bằng C/C++. Nó cũng nằm trong bộ trình dịch mã nguồn mở MinGW. Chương trình IDE này được viết bằng ngôn ngữ Delphi.

Dev-C++ là IDE dành cho C/C++. C/C++ là một ngôn ngữ lập trình cực mạnh, có khả năng tương tác cao và thích ứng với các hệ điều hành khác nhau. Hiện nay, có rất nhiều IDE hỗ trợ lập trình C/C++ như Turbo C, Visual C++, ... Và Bloodshed Dev-C++ là IDE Portable đầu tiên.

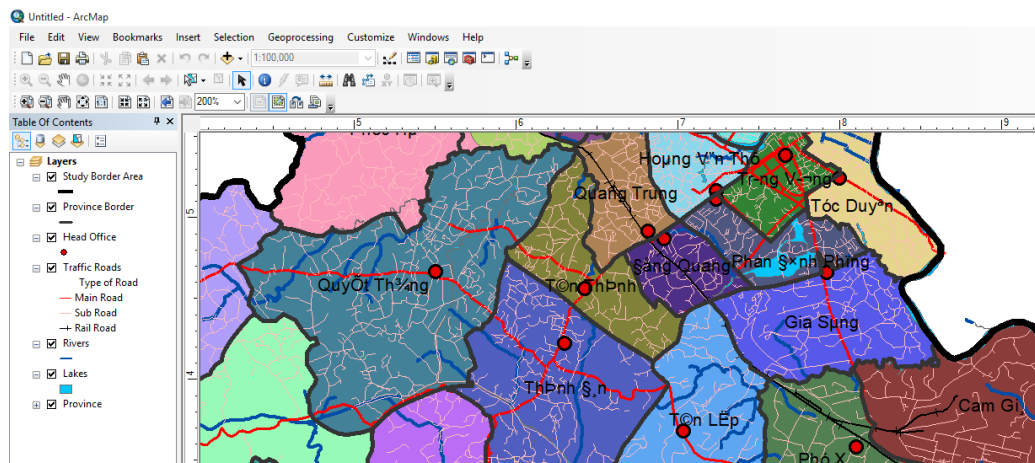


Hình 3.7. Chương trình DEV C

Chương trình hỗ trợ đầy đủ các chức năng mà một IDE chuyên nghiệp, từ việc soạn thảo, trợ giúp đến debug và hỗ trợ thư viện cho các lập trình viên C/C++ (kể cả viết chương trình cho DOS và chương trình cho Windows). Với IDE này, ta dễ dàng tạo ra một chương trình C/C++ một cách chuyên nghiệp, nhanh chóng và đơn giản.

3.3.2. Chương trình thử nghiệm

Chương trình liệt kê trong phần phụ lục của luận văn.



Hình 3.8. Thí dụ chọn các tọa độ trên bản đồ trong ArcGIS

Ứng dụng thực tế:

Hiện nay để xác định vùng lãnh thổ mới: cơ quan, trường học, gia đình hay một lãnh thổ mới... cần xác định tọa độ các điểm để vẽ lên bản đồ số.

Trong khuôn khổ luận văn bản đồ số được xác định: VD như Bản đồ Trường công nghệ thông tin và truyền thông. Xác định tọa độ các điểm xung quanh trường tạo nên đa giác n cạnh.

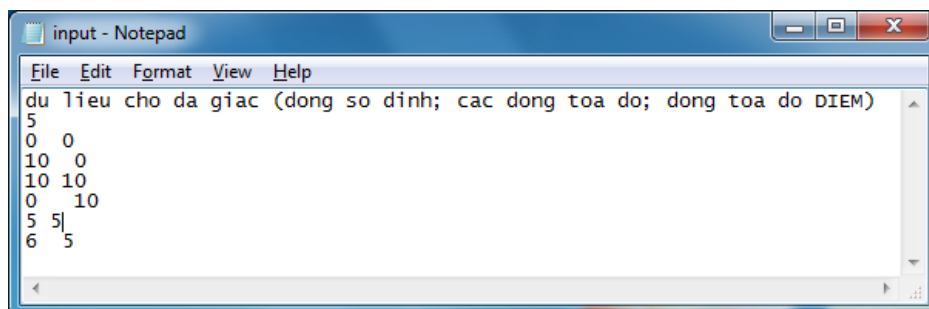
Để xác định một điểm có thuộc đa giác đó có thuộc đa giác hay không giúp cho việc quản lý học sinh, sinh viên, cán bộ giáo viên, nhân viên ... được thực hiện một cách hiệu quả và ứng dụng được các công việc điều hành.

Đầu vào

- 1- Tọa độ các đỉnh của đa giác trên bản đồ, tức các tọa độ trong mặt phẳng;
- 2- Tọa độ của điểm cần kiểm tra.

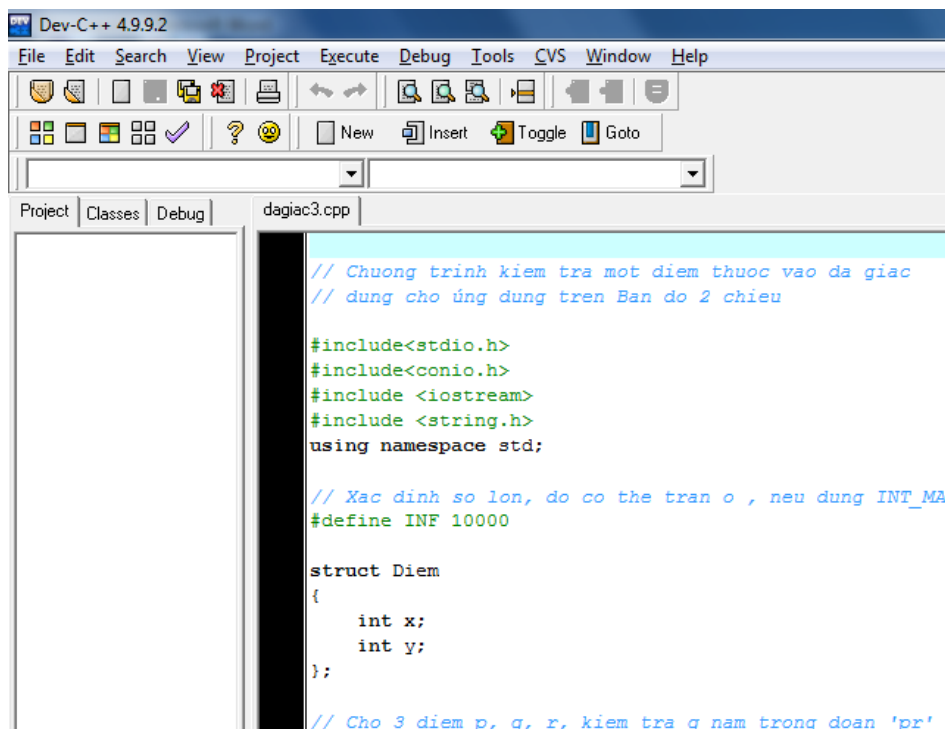
Đầu ra Kết luận về điểm có thuộc diện tích bản đồ, lãnh thổ, mà đa giác đó giới hạn.

Thuật toán: Thử nghiệm theo thuật toán Ray Casting.



```
input - Notepad
File Edit Format View Help
du lieu cho da giac (dong so dinh; cac dong toa do; dong toa do DIEM)
5
0 0
10 0
10 10
0 10
5 5
6 5
```

Hình 3.9. Dữ liệu đầu vào



```
Dev-C++ 4.9.2
File Edit Search View Project Execute Debug Tools CVS Window Help
// Chương trình kiểm tra một điểm thuộc vào đa giác
// dùng cho ứng dụng trên Bản đồ 2 chiều

#include<stdio.h>
#include<conio.h>
#include <iostream>
#include <string.h>
using namespace std;

// Xác định số lớn, do có thể tràn o , nếu dùng INT_MAX
#define INF 10000

struct Diem
{
    int x;
    int y;
};

// Cho 3 điểm p, q, r, kiểm tra q nằm trong đoạn 'pr'
```

Hình 3.10. Chương trình trong DEV C

```
Chương trình kiểm tra Điểm thuộc Đa Giác
Toa do cac diem cua da giac

D[1].x = 0    D[1].y = 0
D[2].x = 10   D[2].y = 0
D[3].x = 10   D[3].y = 10
D[4].x = 0    D[4].y = 10
D[5].x = 5    D[5].y = 5

Toa do diem can kiểm tra :
x= 6 y= 5
Thuoc vào đa giác
```

Hình 3.11. Kết quả đầu ra

3.4. Kết luận

Chương trên đã trình bày các khả năng của phần mềm quản lí bản đồ trong ArcGIS.

Một số thuật toán hình học giới thiệu trong chương có ý nghĩa đối với khả năng ứng dụng trên bản đồ số.

Phần thử nghiệm trong chương trên là thuật toán kiểm tra một điểm thuộc vào đa giác. Thuật toán Ray Casting được chương trình hóa trong môi trường DEV C, cho phép thử nghiệm với đa giác lõm và đa giác lồi.

Luận văn đã bổ sung một số ứng dụng của bài toán xác định điểm thuộc đa giác, trong (i) bản đồ địa chính; (ii) bản đồ số; (iii) vùng lãnh hải; (iv) vùng không phận.

Kết luận

Kết quả đạt được

Sau thời gian thực hiện luận văn tốt nghiệp với đề tài xác định điểm thuộc vào vùng địa lí, tôi đã tự nhận thấy bản thân đạt các kết quả sau:

- Về lí thuyết: nêu các thuật toán hình học, sử dụng (i) trong giảng dạy; (ii) thử nghiệm trên bản đồ mô phỏng;
- Về thực hành, ứng dụng: sử dụng thuật toán Ray Casting để kiểm tra một điểm có thuộc vào đa giác.

Việc ứng dụng các thuật toán hình học trong công tác dạy và học môn hình học trong trường phổ thông có ý nghĩa lớn. Tuy nhiên, do trình độ tin học hóa, mọi người chưa có điều kiện tiếp cận nhiều thuật toán và ứng dụng các thuật toán trong hệ thống tin học khác nhau, các phần mềm được ứng dụng trong thực tế rất đa dạng và hiện đại trên nhiều lĩnh vực. Do vậy việc ứng dụng thuật toán Ray Casting trong hệ thống thông tin địa lí là cần thiết đối với học viên, học viên có thể áp dụng cách “xác định một điểm thuộc vùng đa giác ...” để áp cho cơ quan, địa phương mình. Nên đây là cơ hội để ứng dụng công nghệ thông tin.

Trên nền của hệ thống thông tin địa lí, luận văn đã thể hiện các thuật toán vẽ các đường hình học cơ bản, như ellipse, đường tròn, đa giác...

Chương 1 của luận văn đã đề cập một số khái niệm về hình học, vai trò của hình học đối với phát triển tư duy.

Các bài toán về hình học phẳng, đặc biệt là bài toán đối với tam giác, có ý nghĩa trong chương trình môn toán. Trong chương sau, bài toán về đa giác sẽ tạo nhiều ứng dụng hơn nữa.

Chương 2 trình bày những khả năng ứng dụng bài toán hình học trong thực tế. Ngoài chức năng tạo trực quan, các thuật toán hình học được ứng dụng trong hệ thống bản đồ.

Trình bày cụ thể các thuật toán hình học liên quan đến bài toán “Xác định một điểm thuộc vùng đa giác...” so sánh và ứng dụng trong bản đồ số

Phần mềm ArcGIS của hãng ESRI được luận văn nêu, để so sánh thể hiện ứng dụng của bài toán hình học. Đầu chương là một số khái niệm về phần tử hình học, đã được sử dụng trong hệ thống thông tin địa lí.

Chương 3 của luận văn đã thực hiện trình bày các khả năng của phần mềm quản lí bản đồ.

Một số thuật toán hình học giới thiệu trong chương có ý nghĩa đối với việc học tập, để tạo khả năng ứng dụng trên bản đồ số.

Phần thử nghiệm trong chương trên là thuật toán kiểm tra một điểm thuộc vào đa giác. Thuật toán Ray Casting được chương trình hóa trong môi trường DEV C, cho phép thử nghiệm với đa giác lồi và đa giác lõm.

Phương hướng tiếp tục

Tuy nhiên, những kết quả đạt được mới là bước đầu. Tôi nhận thấy mình cần tiếp tục tìm hiểu nhiều hơn về lí thuyết, và thử nghiệm các ứng dụng, nhằm áp dụng các kiến thức hình học vào thực tế.

Cụ thể cần:

- Tiếp tục tìm hiểu về hệ thống thông tin địa lí, các thuật toán hình học áp dụng cho hệ thống thông tin địa lí;
- Hoàn thiện phần mềm, để có thể sử dụng trong một số công việc .

Tài liệu tham khảo

Tiếng Việt

- [1]. Đặng Văn Đức, Hệ thống thông tin địa lí GIS, NXB. Khoa học Kỹ thuật, 2001
- [2]. Đỗ Trung Tuấn, Hệ thống đa phương tiện, Nxb. Đại học Quốc gia Hà Nội, 2012
- [3]. Euclid, Cơ sở của hình học, NXB. Tri thức, 2015

Tiếng Anh

- [4]. Daniel Weiskopf (2006). GPU-Based Interactive Visualization Techniques. Springer Science & Business Media. p. 21. ISBN 978-3-540-33263-3.
- [5]. Esri Press, The ArcGIS imagery book, new view, new vision, Ed. ESRI, 2016
- [6]. Kiselev, Hình học phẳng, NXB. Thế giới, 2015
- [7]. Roth, Scott D., "Ray Casting for Modeling Solids", Computer Graphics and Image Processing, 18 (2): 109–144, doi:10.1016/0146-664X(82)90169-1
- [8]. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, An introduction to Algorithm, Ed. MIT, 2009

Phụ lục

Chương trình kiểm tra một điểm thuộc đa giác, theo thuật toán Ray Casting

```
// Chuong trinh kiem tra mot diem thuoc vao da giac
// dung cho ung dung tren Ban do 2 chieu

#include<stdio.h>
#include<conio.h>
#include <iostream>
#include <string.h>
using namespace std;

// Xac dinh so lon, do co the tran o, neu dung INT_MAX
#define INF 10000

struct Diem
{
    int x;
    int y;
};

// Cho 3 diem p, q, r, kiem tra q nam trong doan 'pr'

bool onSegment(Diem p, Diem q, Diem r)
{
    if (q.x <= max(p.x, r.x) && q.x >= min(p.x, r.x) &&
        q.y <= max(p.y, r.y) && q.y >= min(p.y, r.y))
        return true;
    return false;
}

// Phat hien huong cua bo ba thu tu p, q, r).
// Ket qua (0): thang hang; (1) theo chieu kim dong ho; (2) nguoc lai

int orientation(Diem p, Diem q, Diem r)
{
    int val = (q.y - p.y) * (r.x - q.x) -
        (q.x - p.x) * (r.y - q.y);

    if (val == 0) return 0; // thang hang
    return (val > 0)? 1: 2; // theo chieu kim dong ho, hoac nguoc lai
}

// Ham cho gia tri DUNG neu doan 'p1q1' cat doan 'p2q2'

bool doIntersect(Diem p1, Diem q1, Diem p2, Diem q2)
{
    // Tim 4 huong can thiet doi voi truong hop tong quat va truong hop
    dac biet

    int o1 = orientation(p1, q1, p2);
    int o2 = orientation(p1, q1, q2);
    int o3 = orientation(p2, q2, p1);
    int o4 = orientation(p2, q2, q1);

    // Truong hop tong quat
    if (o1 != o2 && o3 != o4)
```

```

return true;

// Truong hop dac biet: p1, q1 va p2 la thang hang, va p2 thuoc doan
p1q1
if (o1 == 0 && onSegment(p1, p2, q1)) return true;

// p1, q1 va p2 thang hang, va q2 thuoc doan p1q1
if (o2 == 0 && onSegment(p1, q2, q1)) return true;

// p2, q2 va p1 thang hang, va p1 thuoc doan p2q2
if (o3 == 0 && onSegment(p2, p1, q2)) return true;

// p2, q2 va q1 thang hang, va q1 thuoc doan p2q2
if (o4 == 0 && onSegment(p2, q1, q2)) return true;

return false; // Không rơi vào các trường hợp trên
}

// DUNG neu diem p thuoc vao da giac dagiac[] co n dinh
bool isInside(Diem dagiac[], int n, Diem p)
{
// Da giac can co it nhat 3 dinh
if (n < 3) return false;

// tao diem doi voi doan tu p den vo cung
Diem daukia = {INF, p.y};

// Dem giao diem cua doan tren, voi các canh cua da giac
int count = 0, i = 0;
do
{
int next = (i+1)%n;

// Kiem tra doan thang tu 'p' den 'daukia' cat voi doan tu
'dagiac[i]' den 'dagiac[next]'
if (doIntersect(dagiac[i], dagiac[next], p, daukia))
{
// Neu diem 'p' thang hang voi doan 'i-next', can kiem tra no nam
trong doan ?
// Neu no thuoc, DUNG; nguoc lai SAI

if (orientation(dagiac[i], p, dagiac[next]) == 0)
return onSegment(dagiac[i], p, dagiac[next]);

count++;
}
i = next;
} while (i != 0);

// DUNG neu so dem la LE; nguoc lai SAI
return count&1; // Giong nhu (count%2 == 1)
}

// Chuong trinh chinh
int main()
{
Diem dg[100];
Diem p;

```



```

// doc du lieu tu file
//Khai bao con tro tep

FILE *f = fopen("input.txt","r");
char s[255];
int i, n;

fgets(s,100,f);
fscanf (f,"%d",&n);
//printf ("-----%s-----%d", s, n);

printf ("\nChuong trinh kiem tra Diem thuoc Da GIAC\n Toa do cac
diem cua da giac\n");

for (i=1; i <= n; i++)
{
fscanf (f, "%d%d", &dg[i-1].x, &dg[i-1].y);
printf ("\nD[%d].x = %3d D[%d].y = %3d",i, dg[i-1].x, i, dg[i-1].y);
}
// fgets(s,100,f);
fscanf (f,"%d%d",&p.x, &p.y);
printf ("\n\nToa do diem can kiem tra:\nX= %5d Y=%5d", p.x, p.y);

//Dong con tro tep
fclose(f);

isInside(dg, n, p)? cout << "\nThuoc vao da giac \n": cout <<
"\nKhong thuoc vao da giac \n";

getch();

return 0;
}

```

Chương trình cho thuật toán DDA

```

Procedure DDA ( x1, y1, x2, y2, color: integer );
Var dx, dy, step: integer;
X_inc, y_inc, x, y: real;
Begin
dx:=x2-x1;

dy:=y2-y1;
if abs(dx)>abs(dy) then steps:=abs(dx)
else steps:=abs(dy);
x_inc:=dx/steps;
y_inc:=dy/steps;
x:=x1;
y:=y1;
putpixel(round(x),round(y), color);
for k:=1 to steps do
begin
x:=x+x_inc;

y:=y+y_inc;

putpixel(round(x),round(y), color);
end;

```

```
end;
```

Chương trình cho thuật toán Bresenham

```
Procedure Bres_Line (x1,y1,x2,y2: integer);
  Var dx, dy, x, y, P, const1, const2: integer; Begin
  dx:= x2 - x1;
  dy:= y2 - y1;
  P:= 2*dy - dx;
  Const1:= 2*dy;
  const2:= 2*(dy - dx);
  x:= x1; y:=y1;
  Putpixel ( x, y, Color);
  while (x < x-2 ) do begin
  x:= x +1;
  if (P < 0) then P:= P + const1
  else
  begin
  y:=y +1;
  P:= P + const2
  End;
  putpixel (x, y, color);
  end;
  end;
```

Chương trình thuật toán vẽ đường tròn

```
Procedure Circle (xc, yc, R: integer);
  Var x, y: integer; Procedure DOIXUNG;
  Begin
    putpixel (xc + x, yc +y, color);
  putpixel (xc + x, yc - y, color);
  putpixel (xc - x, yc + y, color);
    putixel (xc -x,yc - y,color) ;
    putpixel (xc + y, yc + x, color);
  putpixel (xc - y, yc + x, color);
  putpixel (xc + y, yc - x, color);
  putpixel (xc - y, yc - x, color);
  end;
  Begin
  For x:= 0 to round(R*Sqrt(2)/2) do
  Begin
  y:= round(Sqrt(R*R - x*x));
  ĐOI XUNG
  End;
  End;
  Chương trình thuật toán MidPoint
  Procedure DTR(xc,yc,mau:integer);
  var x, y, p: integer;
  begin
  x:=0; y:=r;
  p:=1 - r;
  while ( y > x) do begin
  doi_xung;
  if (p < 0) then p:=p+2*x+3
  else begin
  p:=p+2*(x-y)+5;
  y:=y-1;
  end; x:=x+1; end; {while}
  end;
```

Chương trình vẽ đường tròn bằng thuật toán Bresenham

```
Procedure DTR_BRES(xc,yc,r,mau: integer);
var x,y,p:integer;
begin
  x:=0; y:=r;
  p:= 3- 2*r;
  while( x<y) do begin doi_xung;
    if (p<0)thenp:= p+ 4*x + 6
  elsebegin
    p:=p + 4*(x-y)+ 10;
    y:=y-1;
  end; x:=x+1; end;{while}
end;
```

Chương trình thuật toán vẽ đường ellipse

```
Procedure Ellipse(xc,yc,a,b: integer);
varx,y: integer;
  z1, z2, P: real;
  Procedure dx;
  Begin
  putpixel (xc + x, yc +y, color);

  putpixel (xc - x, yc + y, color); putpixel (xc + x, yc - y, color);
  putpixel (xc - x, yc- y, color);
  end;
  begin
  x:=0 y:=b;
  z1:= (b*b)/(a*a);

  z2:= 1/ z1;
  P:= 2*z1 - 2*b +1;
  while (z1* (x/y) ≤ 1) do
    begin
      dx;
      if P < 0 then P:= P + 2*z1*(2*x+3)
      else begin
        P:= P + 2*z1*(2*x+3) + 4*(1-y);
        y:= y -1;
      end;
      x:= x+1;
    end;
    x:=a; y:= 0;
    P:= 2*z2 - 2*a +1;
    while (z2* (y/x) < 1) do
      begin
        dx;
        if P < 0 then P:= P + 2*z2*(2*y+3)
        else begin
          P:= P + 2*z2*(2*y+3) + 4*(1-x);
          x:= x -1;
        end;
        y:= y +1;
      end;
    end;
```