



BÀI TẬP LỚP NHÀ

Đồ Họa Máy Tính



MARCH 1, 2024

VISION GROUP
21TGMT

Table of Contents

1. “Thông minh hóa qui trình sản xuất và quản lý xh trên thế giới ở diện rộng với nền tảng chuyển đổi số, trong đó AI giữ vai trò cốt lõi”, giải thích.	5
2. Sự khác biệt giữa mắt người và thiết bị số thu nhận hình ảnh?	6
3. Nền tảng chuyển đổi số là gì?	6
4. Quy trình sản xuất và quy trình quản lý xã hội, dữ liệu thị giác tham gia ở khúc nào?6	
5. Trình bày thách thức về lỗ hổng ngữ nghĩa (Semantic gap).....	6
6. Tại sao Humanoid Robot cần phải có hình người?	7
7. Phân biệt 3 môn: Đồ họa máy tính, xử lý ảnh và thị giác máy tính	7
8. Phân biệt AR và VR. Tại sao chúng ta cần nó?.....	8
9. So sánh các tác vụ sau: Detection, Recognition, Classification, Tracking, Retrieval, Counting, Generating	8
10. Generating khác các tác vụ trước ở điểm nào?	9
11. Vì sao không dùng dữ liệu ảnh mà dùng dữ liệu 3D points cloud?	10
12. Thông minh là gì? Giải thích câu nói: "Intelligence is the ability to adapt to change" (Stephen HawKing)	10
13. Thông tuệ là gì? Khác gì hơn với thông minh?.....	10
14. Giải thích sơ đồ Training a Neural Network?	10
15. AI giúp giải quyết thách thức semantic gap như thế nào? (Giải quyết bài toán rất tốt chỗ nào? - Dữ liệu máy lưu trữ là dữ liệu thô, là đám mây điểm có giá trị độ sáng mà máy lưu.)	10
16. Mục đích của đồ họa máy tính? Thành phần cơ bản của 1 hệ thống đồ họa máy tính?	11
17. Giải thích các ứng dụng trong slide 5 của Powerpoint đầu tiên?	12

18.	Mục đích của đồ họa máy tính mô phỏng ảnh thế giới thực 3 chiều, nhưng màn hình chỉ có 2 chiều?	13
19.	Số màu có thể hiện được trên máy tính? Liệu như thế đã đủ chưa?	13
20.	Đối tượng 3 chiều phân rã thành đối tượng 2 chiều cơ bản như thế nào?.....	13
21.	Làm thế nào để mô phỏng thế giới thực trong một màn hình nhỏ?	14
22.	Trình bày phương pháp vẽ đoạn thẳng ($y = ax + b$) m tùy ý dựa trên phương pháp với $0 < m < 1$	14
23.	Sử dụng Bresenham's Line để tạo đường thẳng với A(20, 10) và B(30, 18). 15	
24.	Vẽ ngôi sao 5 cánh đều với tâm C, r1, r2. Tính tự động số viên gạch trong bán kính pi để viền hồ hình sao (c, r(m), r2(m))	16
25.	Trình bày cách vẽ hình tròn.....	17
26.	Có nguyên lý gì chi phối việc vẽ một đoạn thẳng?	18
27.	Tại sao chúng ta cần đường tròn? Và ta có thể dùng đường thẳng vẽ đường tròn được không?.....	18
28.	Tại sao đường thẳng, cung tròn và cung eclipse không thể vẽ mọi thứ?	18
29.	Tìm cách vẽ đường eclipse.....	18
30.	Tại sao không vẽ trực tiếp đường cong Bezier mà phải qua các điểm bên ngoài?	20
31.	Đường cong Beizer đi qua 2 đầu mút có lợi gì?.....	20
32.	Cho $n = 3$ thì đường cong Bezier bậc 3 có hình dạng như thế nào?	20
33.	Vẽ đường cong Bezier bậc 3 trên lưới tọa độ nguyên.....	21
34.	Vẽ đường cong Bezier bậc 3 dựa vào phương thức De Casteljau.....	22
35.	Chứng minh phương thức của De Casteljau trùng với đường cong của Bezier:	23
36.	Tại sao đường cong Bezier ứng dụng chỉ dùng đến bậc 3?.....	23
37.	Đường tròn và elip có thuộc lớp C2 hay không?	23

38.	Vì sao tính chất đi qua 2 điểm đầu mút của đường cong Bezier thỏa tính chất của lớp C0.....	24
39.	Tại sao tính chất 2 của đường cong Bezier (tiếp xúc 2 tiếp tuyến đầu cuối) đảm bảo thuộc lớp C1.....	24
40.	Vế trái và vế phải trong tính chất Affine invariance property suy ra được gì?	24
41.	Viết giải thuật vẽ đường cong bậc 3 dùng phương pháp De Casteljau	24
42.	Vì sao theo Jordan Curve Theorem, nó chỉ tô từ lẻ tới chẵn mà không tô từ chẵn tới lẻ?	25
43.	Làm sao để biết đỉnh nào là đỉnh cực trị? Nếu gặp đỉnh không phải là cực trị thì phải giải quyết như thế nào? (Scan-line Polygon-Fill Method).....	25
44.	Cho đa giác n cạnh (n bất kỳ). Tính diện tích và chu vi.....	26
45.	Tìm phép biến đổi sao cho ảnh không bị biến dạng sau khi biến đổi	26
46.	Tìm phép biến đổi WTV?.....	26
47.	Biến đổi S_x , S_y , trong công thức WTV sao cho ảnh đưa vào viewport vẫn giữ đúng tỉ lệ sau khi biến đổi?.....	27
48.	Ngay cả khi có WTV và VTW vẫn chưa thể thiết kế được. Tại sao?	28
49.	Tính tọa độ của các đỉnh trong tọa độ mặt cầu?.....	28
50.	Viết phương trình của mặt Tabsurf.....	28
51.	Viết phương trình toán cho các mặt vẽ (surfaces).....	28
52.	Giải thích vì sao, khi chiếu từ mặt sau, nếu mặt phẳng alpha có tích vô hướng với mặt chiếu > 0 thì mặt alpha đó không nhìn thấy (không được vẽ lên canvas)	29
53.	Xác định khi nào 1 đỉnh nằm trong clipping window, khi nào nằm ngoài clipping window.	30
54.	Viết giải thuật line clipping	30

1. “Thông minh hóa quy trình sản xuất và quản lý xã hội trên thế giới ở diện rộng với nền tảng chuyển đổi số, trong đó AI giữ vai trò cốt lõi”, giải thích.

Trí tuệ nhân tạo (AI) là một lĩnh vực của khoa học máy tính liên quan đến việc tạo ra các hệ thống máy tính có thể thực hiện các nhiệm vụ thông minh mà con người thường thực hiện. AI đã được ứng dụng trong nhiều lĩnh vực khác nhau, từ sản xuất đến dịch vụ, và ngày càng đóng vai trò quan trọng trong quá trình chuyển đổi số.

Trong quá trình chuyển đổi số, AI giữ vai trò cốt lõi bởi những lý do sau:

AI có khả năng tự động hóa các quy trình: AI có thể tự động hóa các quy trình thủ công, từ đơn giản đến phức tạp, giúp giảm chi phí, tăng năng suất và hiệu quả.

AI có khả năng phân tích dữ liệu: AI có thể phân tích dữ liệu lớn một cách nhanh chóng và hiệu quả, giúp các doanh nghiệp đưa ra các quyết định chính xác hơn.

AI có khả năng học hỏi và thích ứng: AI có thể học hỏi và thích ứng với môi trường thay đổi, giúp các doanh nghiệp thích ứng với những thay đổi của thị trường.

Cụ thể, trong quá trình chuyển đổi số ở các lĩnh vực sản xuất và quản lý xã hội, AI được ứng dụng trong nhiều khía cạnh như:

Trong sản xuất: AI được ứng dụng trong tự động hóa sản xuất, kiểm soát chất lượng, quản lý chuỗi cung ứng,... giúp nâng cao năng suất, hiệu quả và giảm chi phí sản xuất.

Trong quản lý xã hội: AI được ứng dụng trong các lĩnh vực như y tế, giáo dục, tài chính, giao thông,... giúp cải thiện chất lượng cuộc sống của người dân.

Ví dụ, trong lĩnh vực sản xuất, AI được sử dụng để tự động hóa các quy trình sản xuất, chẳng hạn như lắp ráp, kiểm tra chất lượng, đóng gói,... AI cũng được sử dụng để tối ưu hóa các quy trình sản xuất, chẳng hạn như lập kế hoạch sản xuất, quản lý kho bãi,... Nhờ AI, các doanh nghiệp sản xuất có thể tăng năng suất, hiệu quả và giảm chi phí sản xuất.

Trong lĩnh vực quản lý xã hội, AI được sử dụng để phát triển các ứng dụng chăm sóc sức khỏe, giáo dục, tài chính, giao thông,... AI giúp cải thiện chất lượng cuộc

sống của người dân, chẳng hạn như cung cấp dịch vụ chăm sóc sức khỏe tốt hơn, giáo dục chất lượng cao hơn, tài chính dễ tiếp cận hơn, giao thông thuận tiện hơn,...

2. Sự khác biệt giữa mắt người và thiết bị số thu nhận hình ảnh?

Mắt người liên kết với não, dây thần kinh thị giác dẫn đến bộ phận xử lý trên não và điều tiết cách thu nhận hình ảnh của mắt.

Độ phân giải: mắt người thu nhận được 1 độ phân giải cố định, thiết bị kỹ thuật số có thể tăng độ phân giải theo tiên bộ kỹ thuật.

Camera số làm mất đi thông tin chiều sâu, chiếu hình ảnh lên mặt phẳng 2 chiều -> bài toán suy ra được giá trị thực bên ngoài từ hình ảnh ghi nhận được

Camera chỉ lưu nhận dữ liệu các điểm ảnh (pixel); não người thu nhận dữ liệu và hiểu được hình ảnh thu nhận (image understanding).

3. Nền tảng chuyển đổi số là gì?

Nó không chỉ là thay đổi tất cả thông tin về dạng số (số hóa), nó còn là thay đổi cách vận hành và áp dụng ứng dụng thiết bị điện tử để sử dụng những thông tin dạng số đó sao cho hiệu quả.

4. Quy trình sản xuất và quy trình quản lý xã hội, dữ liệu thị giác tham gia ở khúc nào?

Trong quy trình sản xuất, dữ liệu thị giác tham gia ở khúc sản xuất, gia công và kiểm định chất lượng của sản phẩm.

Trong quản lý xã hội, dữ liệu thị giác tham gia ở nhiều ở nhiều khúc và điển hình nhất là xác định danh tính của một công dân.

5. Trình bày thách thức về lỗ hổng ngữ nghĩa (Semantic gap)

Lỗ hổng ngữ nghĩa là chỉ ra những đặc trưng khác biệt về các cách mô tả về một đồ vật bằng những ‘ngôn ngữ’ khác nhau. Điển hình dễ thấy nhất là ở “Image Analysis”, là sự khác biệt về ngữ nghĩa giữa hình ảnh lưu được và hình ảnh thật nên cần thu hẹp khoảng cách này lại.

Khoảng cách này hiện tại được thu hẹp lại rất nhỏ, gần với thực tế nhờ kỹ thuật Deep Learning.

6. Tại sao Humanoid Robot cần phải có hình người?

Humanoid robot được tạo ra để mô phỏng lại cấu trúc cơ thể và hành vi của con người, giúp mang lại những hiểu biết hơn về con người, phục vụ nghiên cứu trong một số lĩnh vực khoa học, nghiên cứu nhận thức, hành vi.

Bên cạnh đó, humanoid robot có thể tương tác với con người dễ dàng hơn, thuận lợi hơn cho viễn cảnh tương lai khi con người sống và làm việc trong một môi trường tương tác thường xuyên với các robot, từ đó nhận được nhiều sự ủng hộ hơn từ cộng đồng công nghệ. Vì thế, con người có xu hướng dễ tương tác với những robot này hơn các dạng robot khác.

Ví dụ: Trong ngành y học, nghiên cứu hoặc là ngành giải trí,...

7. Phân biệt 3 môn: Đồ họa máy tính, xử lý ảnh và thị giác máy tính

Các tiêu chí so sánh: Vai trò, Input, Output

	Đồ họa máy tính	Xử lý ảnh	Thị giác máy tính
Vai trò	Mô phỏng VR	Tăng cường chất lượng thô tạo tiền đề cho các ứng dụng	Tăng cường khả năng hiểu ảnh / video cho máy tính
Input	Các điểm 2D, 3D và Object	Hình ảnh	Hình ảnh
Output	Các đường cong 2D, 3D hoặc là các object 3D kèm theo animation và illumination	Hình ảnh chất lượng cao hơn	Ngữ nghĩa của bức ảnh
⇒ Ba môn học này có mối quan hệ chặt chẽ với nhau và bổ sung cho nhau, tạo nên một kiến thức nền tảng vững chắc cho sinh viên, giúp sinh viên có thể tiếp			

tục học tập và nghiên cứu ở các cấp độ cao hơn. Việc bỏ một trong ba môn học này sẽ dẫn đến việc kiến thức nền tảng của sinh viên không được hoàn thiện, ảnh hưởng đến khả năng học tập và nghiên cứu của sinh viên ở các cấp độ cao hơn.

8. Phân biệt AR và VR. Tại sao chúng ta cần nó?

	AR	VR
Phân biệt	Có phụ thuộc vào thế giới thực	Hoàn toàn là ảo, không phụ thuộc vào thế giới thực
Tại sao	Tăng trải nghiệm ở thế giới thực lẫn ảo	Tăng trải nghiệm với thế giới ảo

Hiện thực hiện tại không đủ.

Hiện thực ảo: phần ảo (virtual) chiếm đa số, hướng vào môi trường ảo; hiện thực tăng cường: đa phần vẫn ở hiện thực thông thường, chỉ bổ sung thêm 1 số phần ảo.

AR tăng cường trải nghiệm bằng cách thêm vào các yếu tố ảo và được sử dụng nhiều trong các thiết bị di động để thay đổi cách Thế giới thực và các hình ảnh kỹ thuật số, đồ họa giao nhau và tương tác lẫn nhau để thay đổi cách Thế giới thực và các hình ảnh kỹ thuật số, đồ họa giao nhau và tương tác lẫn nhau.

Thế giới thực của VR, ngược lại, hoàn toàn được tạo ra và điều khiển bởi máy tính và được vận chuyển tới người sử dụng thông qua các thiết bị giúp người dùng kết nối với thực tế ảo và cho phép họ kiểm soát và định vị hành động của mình trong môi trường mô phỏng.

9. So sánh các tác vụ sau: Detection, Recognition, Classification, Tracking, Retrieval, Counting, Generating

	Vai trò	Input	Output	Ví dụ
Detection	Tìm vị trí và gán class ID	Hình ảnh	Class ID của input và bounding box	Ảnh có nhiều mặt người, thì nó nhận diện

			của các vật trong input	được mặt người (nói chung) ở đâu
Recognition	Đi tìm cụ thể ID của object	Hình ảnh	ID object cụ thể	Nhìn mặt thì sẽ biết mặt đó là ai
Classification	Biết được class ID của cả hình ảnh	Hình ảnh	Class ID của ảnh	Đưa ảnh xe thì trả về ID ảnh là “xe”, đưa ảnh mèo thì trả về ID ảnh là “mèo”
Tracking	Biết được vị trí của object đang được theo dõi	Chuỗi hình ảnh	Bounding box của object cần theo dõi	
Retrieval	Trả về một danh sách có thứ tự của các ảnh	Ảnh Query (Q)	Chuỗi hình ảnh được xếp theo thứ tự gần giống Q nhất tới khác Q nhất	
Counting	Số lượng object	Hình ảnh	Một con số	
Generating	Tạo ra một Output mới	Chữ, âm thanh,...	Hình ảnh	

10. Generating khác các tác vụ trước ở điểm nào?

Các tác vụ trước cần cho trước ảnh; generating liên kết được các domain dữ liệu khác nhau (text, voice, image, brain signal, etc)

Generating sinh ra được cái mới; các tác vụ trước chỉ lấy lại những gì đang có

11. Vì sao không dùng dữ liệu ảnh mà dùng dữ liệu 3D points cloud?

Máy đọc dữ liệu, đọc được độ phản xạ (reflection) bên trong, thông số về mật độ điểm (density), dựa vào đó mà máy có thể phân lớp được.

12. Thông minh là gì? Giải thích câu nói: "Intelligence is the ability to adapt to change" (Stephen HawKing)

Thông minh là khả năng nhận biết bản chất của sự việc hiện hữu trong 1 gian nhanh nhất.

Intelligence is the desire to learn and do more of what works better and faster and always works in progress. Adaptability is one of the key measures of intelligence.

13. Thông tuệ là gì? Khác gì hơn với thông minh?

Thông tuệ là biết dùng trí thông minh hiệu quả, vào thực tế, ngữ cảnh nhất định; có trải nghiệm bền vững (wisdom)

Tính thông minh có tính trách nhiệm (khôn ngoan), được sử dụng có trách nhiệm --
> Mong muốn AI có trách nhiệm --> AI là trí thông minh nhân tạo --> vấn đề về ethics

14. Giải thích sơ đồ Training a Neural Network?

Kiến trúc mạng tạo trước, theo kiến trúc định sẵn đưa ra output. Khi học, đánh giá output với nhãn thực, tính sai số -> cực tiểu sai số -> lấy đc bộ tham số w -> giải pháp cần tìm ---> Quy về toán (tìm cực tiểu của hàm), phương pháp tối ưu có/ không có ràng buộc

15. AI giúp giải quyết thách thức semantic gap như thế nào? (Giải quyết bài toán rất tốt chỗ nào? - Dữ liệu máy lưu trữ là dữ liệu thô, là đám mây điểm có giá trị độ sáng mà máy lưu.)

Giải quyết Semantic Gap:

AI giải quyết “semantic gap” bằng cách sử dụng các mô hình học máy tiên tiến như mạng nơ-ron nhân tạo (ANN) và các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP), các phương pháp như học sâu (Deep Learning), học chuyển giao (Transfer Learning), và học tăng cường (Reinforcement Learning) cũng được sử dụng rộng rãi. Một cách để đo lường mức độ chính xác của mô hình biểu diễn thông tin trong hình ảnh là sử dụng sai số giảm. Sai số giảm là mức độ khác biệt giữa giá trị thực và giá trị dự đoán của mô hình. Một mô hình biểu diễn thông tin chính xác sẽ có sai số giảm thấp.

Đánh giá mức độ thông minh đầu ra của các hệ thống thị giác:

Tính chính xác: Đầu ra của hệ thống phải chính xác và không có sai sót.

Tính đầy đủ: Đầu ra của hệ thống phải bao gồm tất cả thông tin cần thiết.

Tính kịp thời: Đầu ra của hệ thống phải được cung cấp kịp thời để có thể đưa ra quyết định.

Tính linh hoạt: Đầu ra của hệ thống phải linh hoạt và có thể thích ứng với các tình huống khác nhau.

Tuy nhiên, cần lưu ý rằng các tiêu chí này có thể thay đổi tùy thuộc vào ngữ cảnh cụ thể và mục tiêu của hệ thống thị giác. Ví dụ, trong một ứng dụng giám sát an ninh, tính chính xác và kịp thời có thể quan trọng hơn so với tính đầy đủ. Trong khi đó, trong một ứng dụng phân loại hình ảnh, tính đầy đủ và chính xác có thể được ưu tiên hơn.

16. Mục đích của đồ họa máy tính? Thành phần cơ bản của 1 hệ thống đồ họa máy tính?

Đồ họa máy tính có mục đích làm cho máy tính thực hiện được việc biểu diễn hình ảnh một cách hiệu quả. Ngoài ra, đồ họa máy tính còn có mục đích xử lý những dữ liệu hình ảnh nhận được từ thế giới thực.

Thành phần cơ bản của một hệ thống đồ họa máy tính:

- Đầu vào

- Màn hình
- Các chức năng liên quan tới xử lý dữ liệu hình ảnh (Vẽ, biến đổi hình học, sửa ảnh, hoạt hình, ...)
- Bộ nhớ
- Đầu ra

17. Giải thích các ứng dụng trong slide 5 của Powerpoint đầu tiên?

Đồ họa được ứng dụng rất nhiều trong đời sống của con người chúng ta. Cụ thể:

- Công việc văn phòng, đồ họa máy tính hỗ trợ việc
 - Vẽ các loại biểu đồ khác nhau
 - Cung cấp giao diện làm việc cho người dùng
 - Cung cấp hiệu ứng và giao diện cho việc thuyết trình trong các cuộc họp
 - ...
- Chế bản điện tử (Desktop Publishing): Để có thể xuất bản ra một cuốn sách hoặc một bài báo, những người thiết kế cần một không gian để có thể thử nghiệm các ý tưởng để thiết kế những tài liệu nói trên. Đồ họa máy tính sẽ giúp người thiết kế dễ dàng thử nghiệm nhiều kiểu thiết kế khác nhau từ bố cục trang tới phông chữ và màu sắc.
- CAD-CAM: Đồ họa máy tính giúp tạo ra một môi trường ảo để có thể tạo ra một mẫu sản phẩm (2 chiều hoặc 3 chiều) bất kì trên máy tính. Thêm vào đó, đồ họa máy tính cung cấp các công cụ để người thiết kế tạo ra hoặc chỉnh sửa những thông tin mang tính hình ảnh ở thời gian thực.
- Nghệ thuật: Đồ họa máy tính giúp các họa sĩ hoặc nhà thiết kế tạo ra những tác phẩm nghệ thuật 2D và kể cả 3D một cách dễ dàng, ít tốn kém chi phí và hiệu quả.
- Quy trình kiểm soát: Đồ họa máy tính giúp đơn giản hóa và thực hiện quy trình kiểm soát nhanh hơn, cụ thể là ở bước đo lường kết quả.

- Giải trí: Trong phim ảnh, màn hình xanh (green screen) được sử dụng rất nhiều để có thể tạo dựng background một cách dễ dàng và ít tốn kém chi phí. Và màn hình xanh đó hoạt động được như thế là nhờ vào đồ họa máy tính
- Giáo dục: Đồ họa máy tính có thể được sử dụng để mô phỏng các thí nghiệm khoa học, hoặc là mô phỏng một số mô hình để giúp việc học tập trở nên dễ dàng hơn

18. Mục đích của đồ họa máy tính mô phỏng ảnh thế giới thực 3 chiều, nhưng màn hình chỉ có 2 chiều?

Màn hình chỉ có 2 chiều, nghĩa là chỉ có thể hiển thị các hình ảnh có chiều rộng và chiều cao. Tuy nhiên, đồ họa máy tính có thể sử dụng các kỹ thuật khác nhau để tạo ra ảo giác về chiều sâu và không gian. Các kỹ thuật này bao gồm:

Sử dụng các kỹ thuật phối cảnh: Kỹ thuật phối cảnh là một kỹ thuật toán học được sử dụng để tạo ra ảo giác về chiều sâu và không gian trong hình ảnh. Đồ họa máy tính có thể sử dụng kỹ thuật phối cảnh để tạo ra ảo giác về các vật thể ở xa hơn hoặc gần hơn với màn hình. Có hai loại phép chiếu phối cảnh chính:

Phép chiếu phối cảnh một điểm: Trong phép chiếu phối cảnh một điểm, tất cả các đường thẳng song song đều hội tụ tại một điểm gọi là điểm biến dạng. Điểm biến dạng thường được đặt ở tâm của hình ảnh.

Phép chiếu phối cảnh hai điểm: Trong phép chiếu phối cảnh hai điểm, tất cả các đường thẳng song song nằm ngang đều hội tụ tại hai điểm gọi là điểm biến dạng. Điểm biến dạng thường được đặt ở hai bên của hình ảnh.

19. Số màu có thể hiện được trên máy tính? Liệu như thế đã đủ chưa?

Số lượng màu có thể hiện được là: $256^3 = 16.777.216$ màu

Về vấn đề liệu số lượng màu như trên đã đủ chưa thì câu trả lời chưa thật sự rõ vì máy tính có thể có màu mà ở ngoài tự nhiên không có, nhưng có vài màu tự nhiên mà máy tính không thể mô phỏng lại được

20. Đối tượng 3 chiều phân rã thành đối tượng 2 chiều cơ bản như thế nào?

- Điểm (x, y, z)
- Đường thẳng trong 3 chiều
- Mặt phẳng/ Polygon
- 3D Mesh
- Primitive (box/ sphere/ cylinder/ cone/ loft/ edgesurf/ torus/ revsurf/ ...)

21. Làm thế nào để mô phỏng thế giới thực trong một màn hình nhỏ?

➔ Hardware solution fail. Software solution: ánh xạ tập điểm 3 chiều vào cửa sổ có kích thước nhỏ hơn (scale) -> tất cả điểm đều đc đưa vào màn ảnh. Cần có các phương pháp ánh xạ từ thế giới thực vào màn hình và ngược lại để thiết kế

- Sử dụng phép zoom, pan...
- ViewToWin, WintoView.
- Phép chiếu song song và Perspective

22. Trình bày phương pháp vẽ đoạn thẳng ($y = ax + b$) m tùy ý dựa trên phương pháp với $0 < m < 1$

Phương pháp: Đi tìm hàm **cost** để quyết định điểm cần vẽ tiếp theo dựa vào điểm vẽ hiện tại.

Các bước thực hiện:

B1: Nhập vào điểm bắt đầu và kết thúc, lưu và chấm điểm bắt đầu x_0, y_0

B2: Tính các giá trị hằng

- $\Delta x = |x_1 - x_0|, \Delta y = |y_1 - y_0|$
- $2\Delta y, 2\Delta y - 2\Delta x$ (nếu $|m| < 1$); $2\Delta x, 2\Delta x - 2\Delta y$ (nếu $|m| \geq 1$)
- $p_0 = \begin{cases} 2\Delta y - \Delta x, & |m| < 1 \\ 2\Delta x - \Delta y, & |m| \geq 1 \end{cases}$
- $stepx = \begin{cases} -1, & x_1 < x_0 \\ 1, & x_1 \geq x_0 \end{cases}$
- $stepy = \begin{cases} -1, & y_1 < y_0 \\ 1, & y_1 \geq y_0 \end{cases}$

B3: $k=0$, loop until $k = \Delta x$ (nếu $|m| < 1$) or $k = \Delta y$ (nếu $|m| \geq 1$)

- Nếu $|m| < 1$:
 - Nếu $p \geq 0$:
 - Điểm tiếp theo là $(x + \text{step}_x, y + \text{step}_y)$
 - Cập nhật giá trị p : $p_{k+1} = p_k + 2\Delta y - 2\Delta x$
 - Nếu $p < 0$:
 - Điểm tiếp theo là $(x + \text{step}_x, y)$
 - Cập nhật giá trị p : $p_{k+1} = p_k + 2\Delta y$
- Nếu $|m| \geq 1$:
 - Nếu $p \geq 0$:
 - Điểm tiếp theo là $(x + \text{step}_x, y + \text{step}_y)$
 - Cập nhật giá trị p : $p_{k+1} = p_k + 2\Delta x - 2\Delta y$
 - Nếu $p < 0$:
 - Điểm tiếp theo là $(x, y + \text{step}_y)$
 - Cập nhật giá trị p : $p_{k+1} = p_k + 2\Delta x$
- Chấm điểm tiếp theo
- $k += 1$

23. Sử dụng Bresenham's Line để tạo đường thẳng với A(20, 10) và B(30, 18)

Hằng số:

$$\Delta x = 30 - 20 = 10$$

$$\Delta y = 18 - 10 = 8 \Rightarrow 2\Delta y = 16$$

$$2\Delta y - 2\Delta x = 16 - 20 = -4$$

$$p_0 = 2\Delta y - \Delta x = 6$$

$$(x_0, y_0) = (20, 10)$$

k	(x_k, y_k)	p_k	(x_{k+1}, y_{k+1})	p_{k+1}
0	(20, 10)	6	(21, 11)	2
1	(21, 11)	2	(22, 12)	-2
2	(22, 12)	-2	(23, 12)	14

3	(23, 12)	14	(24, 13)	10
4	(24, 13)	10	(25, 14)	6
5	(25, 14)	6	(26, 15)	2
6	(26, 15)	2	(27, 16)	-2
7	(27, 16)	-2	(28, 16)	14
8	(28, 16)	14	(29, 17)	10
9	(29, 17)	10	(30, 18)	6
10	(30, 18)			

24. Vẽ ngôi sao 5 cánh đều với tâm C, r1, r2. Tính tự động số viên gạch trong bán kính pi để viền hồ hình sao (c, r(m), r2(m))

Input: Tâm ngôi sao C(xC, yC), bán kính r1, r2.

Output: Một ngôi sao 5 cánh đều với tâm C(xC, yC) với bán kính r1, r2.

Bước 1: Tính các hằng số sau:

$$\alpha = 360/5 = 72$$

$$d = r1 + r2$$

$$r = (r1 - r2)/2$$

Bước 2: Vẽ 5 điểm đầu mút của ngôi sao:

Vẽ điểm (xC, yC + r1)

Quay góc α , vẽ điểm (xC - r, yC + r1 - r)

Quay góc α , vẽ điểm (xC - r, yC - r)

Quay góc α , vẽ điểm (xC + r, yC - r + r)

Quay góc α , vẽ điểm (xC, yC + r1)

Bước 3: Vẽ các cạnh của ngôi sao:

Vẽ đoạn thẳng nối 2 điểm đầu mút liên kề, sử dụng các hằng số d và r.

Bước 4: Tính tự động số viên gạch trong bán kính pi để viền hồ hình sao:

Tính bán kính đường tròn nội tiếp ngôi sao:

$$r = (d^2 - r_1^2 - r_2^2)^{(1/2)}$$

Tính bán kính bán kính pi:

$$r_{pi} = 1.1r$$

Tính số viên gạch trong bán kính pi:

$$n = (2 * r_{pi})^2 / (k * s)$$

Trong đó:

k là kích thước viên gạch

s là khoảng cách giữa các viên gạch

25. Trình bày cách vẽ hình tròn

Input: Bán kính **r**, tâm đường tròn **C(x_C, y_C)**

Bước 1: Vẽ điểm $(x_0, y_0) = (0, r)$

Bước 2: Tính các hằng số sau: $2x_0, 2y_0, p_0 = 1 - r$

Bước 3: Cho $k = 0$

Bước 4:

- Nếu $p_k < 0$ thì điểm tiếp theo cần vẽ là $(x = x_k + 1, y = y_k)$ và cập nhật
 - $2x_{k+1} = 2x_k + 2$
 - $p_{k+1} = p_k + 2 \cdot x_{k+1} + 1$
- Nếu $p_k \geq 0$ thì điểm tiếp theo cần vẽ là $(x = x_k + 1, y = y_k - 1)$ và cập nhật
 - $2x_{k+1} = 2x_k + 2$
 - $2y_{k+1} = 2y_k - 2$
 - $p_{k+1} = p_k + 2 \cdot x_{k+1} - 2y_{k+1} + 1$

Bước 5: Từ điểm tiếp theo cần vẽ, xác định và vẽ 7 điểm còn lại đối xứng qua các trục $x = 0$, $y = 0$, $y = x$, $y = -x$

Bước 6: Từ các điểm tìm được ở bước 4 và 5, đi tìm các điểm tương ứng dựa trên (x_C, y_C)

Bước 7: $k = k + 1$

Bước 8: Nếu $x < y$, quay về bước 4. Không thì kết thúc

Output: Một đường tròn tâm $C(x_C, y_C)$ với bán kính r

26. Có nguyên lý gì chi phối việc vẽ một đoạn thẳng?

Đầu tiên, bám theo phương trình đường thẳng

Tiếp theo, tìm cách xấp xỉ tốt nhất trên lưới tọa độ nguyên

27. Tại sao chúng ta cần đường tròn? Và ta có thể dùng đường thẳng vẽ đường tròn được không?

Đường tròn có những đặc tính mà đường thẳng không có:

- Tính chất cong
- Tính đồng đều (Khoảng cách từ tâm tới đường tròn là đều bằng nhau)
- Tính khép kín

Về mặt toán học, chúng ta không thể dùng đường thẳng để vẽ đường tròn.

28. Tại sao đường thẳng, cung tròn và cung eclipse không thể vẽ mọi thứ?

Độ cong của đường tròn không biến thiên liên tục, là hằng số, không đủ uyển chuyển để mô phỏng các đường cong (về mỹ thuật)

Đạo hàm bậc nhất và bậc hai liên quan đến động lực học —> mong muốn có 1 đường cong có đạo hàm bậc nhất, bậc hai biến thiên liên tục để phù hợp hơn với các ứng dụng (về kỹ thuật)

29. Tìm cách vẽ đường eclipse

Input: 2 bán kính của đường ellipse r_x, r_y và tâm $C(x_c, y_c)$

Bước 1: Vẽ điểm $(x_0, y_0) = (0, r_y)$

Bước 2: Tính các hằng số sau: $2.r_y^2.x_0, 2.r_x^2.y_0, p1_0 = r_y^2 - r_x^2.r_y + \frac{1}{4} r_x^2$

Bước 3: Cho $k = 0$

Bước 4:

- Nếu $p1_k < 0$ thì điểm tiếp theo cần vẽ là $(x = x_k + 1, y = y_k)$ và cập nhật
 - $2.r_y^2.x_{k+1} = 2.r_y^2.x_k + 2r_y^2$
 - $p1_{k+1} = p1_k + 2.r_y^2.x_{k+1} + r_y^2$
- Nếu $p1_k \geq 0$ thì điểm tiếp theo cần vẽ là $(x = x_k + 1, y = y_k - 1)$ và cập nhật
 - $2.r_y^2.x_{k+1} = 2.r_y^2.x_k + 2r_y^2$
 - $2.r_x^2.y_{k+1} = 2.r_x^2.y_k - 2 r_x^2$
 - $p1_{k+1} = p1_k + 2.r_y^2.x_{k+1} - 2.r_x^2.y_{k+1} + r_y^2$

Bước 5: Từ điểm mới tìm được, tìm 3 điểm đối xứng đi qua các trục $x = 0, y = 0$

Bước 6: Nếu $2.r_y^2.x < 2.r_x^2.y$ thì $k = k + 1$ và quay về bước 4. Không thì tiếp tục sang bước 7.

Bước 7: Gán $x_{last} = x_k, y_{last} = y_k$

Bước 8: Tính các hằng số sau: $2.r_y^2.x_{last}, 2.r_x^2.y_{last}$

$$p2_0 = r_y^2.(x_{last} + \frac{1}{2})^2 - r_x^2.(y_{last} - 1)^2 - r_x^2.r_y^2$$

Bước 9: $i = 0$

Bước 10:

- Nếu $p2_i \geq 0$ thì điểm tiếp theo cần vẽ là $(x = x_i, y = y_i - 1)$ và cập nhật
 - $2.r_x^2.y_{i+1} = 2.r_x^2.y_i - 2r_x^2$
 - $p2_{i+1} = p2_i - 2.r_x^2.y_{i+1} + r_x^2$
- Nếu $p2_i < 0$ thì điểm tiếp theo cần vẽ là $(x = x_i + 1, y = y_i - 1)$ và cập nhật
 - $2.r_y^2.x_{i+1} = 2.r_y^2.x_i + 2r_y^2$

- $2.r_x^2.y_{i+1} = 2.r_x^2.y_i - 2r_x^2$
- $p_{2i+1} = p_{2i} + 2.r_y^2.x_{i+1} - 2.r_x^2.y_{i+1} + r_x^2$

Bước 11: Từ điểm mới tìm được, tìm 3 điểm đối xứng đi qua các trục $x = 0, y = 0$

Bước 12: Nếu $y_i = 0$ thì chuyển sang bước 13. Không thì $i = i + 1$ và quay về bước 10.

Bước 13: Từ các điểm tìm được ở bước 4, 5, 10, 11, đi tìm các điểm tương ứng dựa trên (x_C, y_C) và kết thúc

Output: Một đường ellipse tâm $C(x_C, y_C)$ với 2 bán kính r_x, r_y

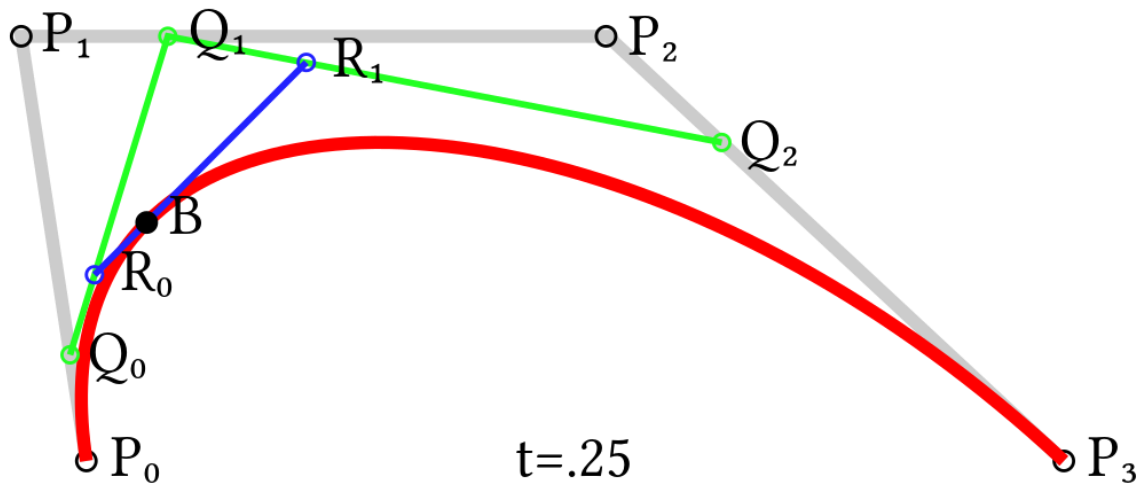
30. Tại sao không vẽ trực tiếp đường cong Bezier mà phải qua các điểm bên ngoài?

Việc vẽ đường cong Bézier thông qua các điểm điều khiển thay vì trực tiếp vì các điểm điều khiển có sự linh hoạt cao hơn, cho phép người thiết kế tinh chỉnh hình dạng của đường cong một cách dễ dàng bằng cách di chuyển hoặc điều chỉnh giá trị của chúng, giúp đơn giản hóa quá trình vẽ hình, thích ứng tốt với sự thay đổi, khiến cho việc điều chỉnh đường cong trở nên hiệu quả mà không làm thay đổi toàn bộ cấu trúc.

31. Đường cong Beizer đi qua 2 đầu mút có lợi gì?

Tránh sau này vẽ đường cong sẽ không bị hở, đứt gãy, làm tiền đề tốt cho những hình sau

32. Cho $n = 3$ thì đường cong Bezier bậc 3 có hình dạng như thế nào?



Nguồn: [Bézier 3 big - Bézier curve - Wikipedia](#)

33. Vẽ đường cong Bezier bậc 3 trên lưới tọa độ nguyên

Input: 4 điểm điều khiển

Output: đường cong Beizier bậc 3

Các bước thực hiện

B1: Tính các giá trị khởi điểm

- $A_{0x} = -(x_0 - 3x_1 + 3x_2 - x_3)$
- $A_{1x} = 3(x_0 - 2x_1 + x_2)$
- $A_{2x} = -3x_0 + 3x_1$
- $M_x = 7|A_{0x}| + 3|A_{1x}| + |A_{2x}|$
- Tính các giá trị tương tự cho $A_{0y}, A_{1y}, A_{2y}, M_y$

B2: Tính các giá trị $dt, dt^2, dt^3, dx, ddx, dddx, dy, ddy, dddy$

- $dt = (\text{Max}(M_x, M_y))^{-1}; dt^2 = dt \cdot dt; dt^3 = dt^2 \cdot dt$
- $dx = A_{0x} \cdot dt^3 + A_{1x} \cdot dt^2 + A_{2x} \cdot dt$
- $ddx = 6A_{0x} \cdot dt^3 + 2A_{1x} \cdot dt^2$
- $dddx = 6A_{0x} \cdot dt^3$
- Tính các giá trị tương tự cho $dy, ddy, dddy$

B3: Chấm điểm đầu tiên cũng là điểm điều khiển đầu tiên (x_0, y_0)

B4: Chấm điểm thứ hai và cập nhật các giá trị

- $x = x_0 + dx; y = y_0 + dy$
- $dx += ddx; dy += ddy$

B5: Tính các điểm tiếp theo trên đường cong

- $step = \text{ROUND}(1/dt)$
- Loop $i = 2; i \leq step; i ++$
 - $x = x_0 + dx; y = y_0 + dy$
 - $dx += ddx; dy += ddy$
 - Chấm điểm ($\text{ROUND}(x), \text{ROUND}(y)$)

B6: Chấm điểm cuối cùng cũng là điểm điều khiển thứ 4 (x_3, y_3)

34. Vẽ đường cong Bezier bậc 3 dựa vào phương thức De Casteljau

```
def bezier_curve_de_casteljau(P0, P1, P2, P3):
```

```
    # Lặp lại cho đến khi đạt được độ chính xác mong muốn
```

```
    def helper(t):
```

```
        if t == 0:
```

```
            return P0
```

```
        elif t == 1:
```

```
            return P3
```

```
        else:
```

```
            return (1 - t) * helper(t / 2) + t * helper((t + 1) / 2)
```

```
    curve_prime = []
```

```
    for t in range(0, 1 + 1):
```

```
curve_prime.append(helper(t))
```

```
return curve_prime
```

35. Chứng minh phương thức của De Casteljau trùng với đường cong của Bezier:

Xét một đường cong Bézier cấp n , được định nghĩa bởi các điểm kiểm soát P_0, P_1, \dots, P_n và tham số $0 \leq t \leq 1$:

$$B(t) = (1-t)^n P_0 + \binom{n}{1} (1-t)^{n-1} t P_1 + \dots + (1-t)^0 t^n P_n$$

Phương pháp của De Casteljau bắt đầu với các điểm kiểm soát P_0, P_1, \dots, P_n và tạo ra các điểm trung gian trên đường cong thông qua việc lặp đi lặp lại việc tạo điểm trung gian giữa các điểm liên kề. Cụ thể, để tạo ra điểm trung gian trên đường cong Bézier, ta có công thức sau:

$$Q_i^j = (1-t)Q_i^{j-1} + tQ_{i+1}^{j-1}$$

Trong đó Q_i^0 là điểm kiểm soát P_i

36. Tại sao đường cong Bezier ứng dụng chỉ dùng đến bậc 3?

Thứ nhất, đường cong Bezier bậc 3 cung cấp được nhiều loại đường cong, đủ để sử dụng trong nhiều ứng dụng.

Thứ hai, bậc Bezier càng cao thì càng có nhiều điểm tác động lên đường cong. Nên khi chỉnh sửa một đường cong theo ý muốn thì việc xử lý nhiều điểm dẫn tới sự phức tạp khá cao.

Thứ ba, bậc Bezier cao sẽ đòi hỏi nhiều công việc tính toán hơn, dẫn tới lượng tài nguyên cần phải được sử dụng nhiều hơn.

37. Đường tròn và elip có thuộc lớp C2 hay không?

➔ Đường tròn và elip đều là đường cong bậc 2, khi đó đạo hàm bậc 2 của đường tròn và elip sẽ là hằng số, nghĩa là không có sự biến thiên liên tục trên đạo hàm bậc hai nên sẽ thiếu đi sự uyển chuyển. Vì thế đường tròn và elip không thuộc

lớp C2 bởi lớp C2 có đặc trưng là sự uốn chuyển trong đường cong có đạo hàm bậc 2 liên tục trên toàn bộ miền xác định.

38. Vì sao tính chất đi qua 2 điểm đầu mút của đường cong Bezier thỏa tính chất của lớp C0

➔ Vì tính chất của lớp C0 là đường cong liên tục và không có sự đứt gãy giữa các đầu mút, do đó, đường cong Bezier được xây dựng đảm bảo đi qua 2 điểm đầu mút sẽ đảm bảo được tính chất này của lớp C0.

39. Tại sao tính chất 2 của đường cong Bezier (tiếp xúc 2 tiếp tuyến đầu cuối) đảm bảo thuộc lớp C1

➔ Vì tính chất của lớp C1 là đường cong có đạo hàm bậc 1 liên tục trên toàn bộ miền xác định. Đường cong Bezier tiếp xúc với cả 2 tiếp tuyến đầu cuối có nghĩa là đường cong có tiếp tuyến xác định và hướng tiếp tuyến liên tục tại các điểm đầu mút đảm bảo rằng đạo hàm bậc 1 của đường cong luôn liên tục, từ đó thỏa mãn tính chất của lớp C1.

40. Vế trái và vế phải trong tính chất Affine invariance property suy ra được gì?

➔ Vế phải: Có $BEZ(k, n)$, phía trước là điểm \rightarrow đường cong Bezier có được từ những điểm điều khiển mới, điểm điều khiển phía trước là kết quả của việc áp phép biến đổi Affine vào những điểm p_k cũ ($A(p_k) + T$).

Vế trái: khi áp phép biến đổi affine vào đường cong Bezier cũ thì sẽ tạo sinh một đường cong mới được tạo dựng từ các điểm điều khiển mới được sinh từ các điểm điều khiển cũ qua phép biến đổi affine.

41. Viết giải thuật vẽ đường cong bậc 3 dùng phương pháp De Casteljau

Input: 4 điểm control a, b, c, d

Output: Đường cong Bezier bậc 3

Bắt đầu

Bước 1: Đặt $t = 0$

Bước 2: Yêu cầu người dùng nhập n

Bước 3: Đặt $Dt = 1/n$

Bước 4: Cho t chạy từ 0 tới 1, bước tăng Dt , lặp các bước sau:

- Bước 4a: Thực hiện phép chia thứ nhất
 - Chia segment p_0p_1 cho t , ta có: $p_{10}(t) = (1 - t) * p_0 + t * p_1$
 - Chia segment p_1p_2 cho t , ta có: $p_{11}(t) = (1 - t) * p_1 + t * p_2$
 - Chia segment p_2p_3 cho t , ta có: $p_{12}(t) = (1 - t) * p_2 + t * p_3$
- Bước 4b: Thực hiện phép chia thứ 2
 - Chia segment $p_{10}p_{11}$ cho t , ta có: $p_{20}(t) = (1 - t) * p_{10} + t * p_{11}$
 - Chia segment $p_{11}p_{12}$ cho t , ta có: $p_{21}(t) = (1 - t) * p_{11} + t * p_{12}$
- Bước 4c: Thực hiện phép chia thứ 3
 - Chia segment $p_{21}p_{22}$ cho t , ta có: $p_{30}(t) = (1 - t) * p_{20} + t * p_{21}$

Bước 3: Nối tập hợp các điểm $p_{30}(t)$ lại

Kết thúc

Có thể dùng cách chia đôi ở t để giảm thiểu độ phức tạp của thuật toán

42. Vì sao theo Jordan Curve Theorem, nó chỉ tô từ lẻ tới chẵn mà không tô từ chẵn tới lẻ?

Theo Jordan Curve Theorem, với một vùng cần tô cho trước. Nếu điểm đó nằm trong vùng cần tô, thì khi kéo dài một đoạn thẳng bắt đầu từ điểm đó (với điều kiện không được đi qua bất kì đỉnh nào của vùng cần tô) và kết thúc ở một điểm nằm ngoài vùng cần tô, số giao điểm của đường thẳng với vùng cần tô sẽ là một số lẻ.

Ngược lại, nếu điểm đó nằm ngoài vùng cần tô thì số giao điểm của đường thẳng với vùng cần tô sẽ là một số chẵn.

Và vì scan line bắt đầu ở ngoài vùng cần tô, nên khi xét từng điểm trên scan line, các điểm đó thỏa với các mô tả ở trên, và sẽ tô từ lẻ tới chẵn thay vì chẵn tới lẻ.

43. Làm sao để biết đỉnh nào là đỉnh cực trị? Nếu gặp đỉnh không phải là cực trị thì phải giải quyết như thế nào? (Scan-line Polygon-Fill Method)

Đối với Scan-line Polygon-Fill Method, để biết đỉnh nào đỉnh cực trị, ta đi khảo sát tung độ ở vị trí liền kề bên trái và bên phải. Nếu tung độ ở hai vị trí này cùng lớn hơn hoặc cùng bé hơn đỉnh đang xét thì đỉnh đó là đỉnh cực trị.

Nếu gặp phải đỉnh không phải đỉnh cực trị thì ta phải rút ngắn một trong 2 đoạn thẳng sao cho 2 đoạn thẳng đó không cắt nhau

44. Cho đa giác n cạnh (n bất kỳ). Tính diện tích và chu vi

Chu vi = Tổng độ dài của n cạnh

Đối với hình đa giác đều n cạnh, thì diện tích sẽ là $A = P \times a / 2$

Trong đó:

A là diện tích

P là chu vi của hình đa giác đều n cạnh

a là khoảng cách từ điểm tới tâm của hình đa giác đều n cạnh

45. Tìm phép biến đổi sao cho ảnh không bị biến dạng sau khi biến đổi

Các phép biến đổi đó là: Phép tịnh tiến, phép xoay, phép đối xứng

46. Tìm phép biến đổi WTV?

Giả sử như sau:

- (x_w, y_w) : Một điểm trên “Window”
- (x_v, y_v) : Điểm tương ứng trên “Viewport”

Ta đi tìm chuẩn hóa các điểm ở trên với:

- Điểm chuẩn hóa trên “Window”: $((x_w - x_{wmin}) / (x_{wmax} - x_{wmin}), (y_w - y_{wmin}) / (y_{wmax} - y_{wmin}))$
- Điểm chuẩn hóa trên “Viewport”: $((x_v - x_{vmin}) / (x_{vmax} - x_{vmin}), (y_v - y_{vmin}) / (y_{vmax} - y_{vmin}))$

Trong đó:

- x_{wmin} : Hoành độ bên trái tận cùng của “Window”
- x_{wmax} : Hoành độ bên phải tận cùng của “Window”
- y_{wmin} : Tung độ bên dưới tận cùng của “Window”
- y_{wmax} : Tung độ bên trên tận cùng của “Window”
- x_{vmin} : Hoành độ bên trái tận cùng của “Viewport”
- x_{vmax} : Hoành độ bên phải tận cùng của “Viewport”
- y_{vmin} : Tung độ bên dưới tận cùng của “Viewport”
- y_{vmax} : Tung độ bên trên tận cùng của “Viewport”

Sau khi có được 2 điểm chuẩn hóa đã được tính ở trên, ta cho hoành độ và tung độ của 2 điểm chuẩn hóa bằng nhau, có nghĩa là:

- $(x_w - x_{wmin}) / (x_{wmax} - x_{wmin}) = (x_v - x_{vmin}) / (x_{vmax} - x_{vmin})$
- $(y_w - y_{wmin}) / (y_{wmax} - y_{wmin}) = (y_v - y_{vmin}) / (y_{vmax} - y_{vmin})$

Từ đó, ta có thể suy ra:

- $x_v = ((x_w - x_{wmin}) * (x_{vmax} - x_{vmin}) / (x_{wmax} - x_{wmin})) + x_{vmin}$
- $y_v = ((y_w - y_{wmin}) * (y_{vmax} - y_{vmin}) / (y_{wmax} - y_{wmin})) + y_{vmin}$

Đặt:

- $S_x = (x_{vmax} - x_{vmin}) / (x_{wmax} - x_{wmin})$: tỉ lệ chiều ngang của viewport so với window
- $S_y = (y_{vmax} - y_{vmin}) / (y_{wmax} - y_{wmin})$: tỉ lệ chiều dọc của viewport so với window

Lúc đó, ta có công thức cuối cùng:

- $x_v = ((x_w - x_{wmin}) * S_x) + x_{vmin}$
- $y_v = ((y_w - y_{wmin}) * S_y) + y_{vmin}$

47. Biến đổi S_x , S_y , trong công thức WTV sao cho ảnh đưa vào viewport vẫn giữ đúng tỉ lệ sau khi biến đổi?

Để ảnh giữ đúng tỉ lệ sau khi biến đổi thì $S_x = S_y = \text{Hằng số}$

Hay $(x_{vmax} - x_{vmin}) / (x_{wmax} - x_{wmin}) = (y_{vmax} - y_{vmin}) / (y_{wmax} - y_{wmin}) = \text{Hằng số}$

48. Ngay cả khi có WTV và VTW vẫn chưa thể thiết kế được. Tại sao?

WTV và VTW chỉ hỗ trợ đưa một hình ảnh lên màn hình, như thế thì chưa đủ cho việc thiết kế vì thiết kế cần nhiều thao tác hơn ngoài việc đưa hình ảnh lên màn hình như là vẽ, xoay, chỉnh sửa, tô màu,...

49. Tính tọa độ của các đỉnh trong tọa độ mặt cầu?

Ta có phương trình mặt cầu trong tọa độ Oxyz là:

$$(x - a)^2 + (y - b)^2 + (z - c)^2 = r^2$$

Trong đó:

Tâm của mặt cầu là **I (a, b, c)** với bán kính mặt cầu là **r**

Vậy để tính tọa độ của các đỉnh trong tọa độ mặt cầu, chỉ cần thay (x, y, z) thỏa mãn phương trình trên.

50. Viết phương trình của mặt Tabsurf

Phương trình tham số của mặt cong Tabsurf được xác định bởi: **$P(u, v) = G(u) + v.n$**

Trong đó:

G(u) là phương trình tham số của đường cong G theo tham số u

n là vector xác định hướng tạo sinh mặt cong

Điều kiện $0 \leq u \leq u_{max}$ và $0 \leq v \leq v_{max}$

51. Viết phương trình toán cho các mặt vẽ (surfaces)

Tabsurf: **$P(u, v) = G(u) + v.n$**

- **G(u)** là phương trình tham số của đường cong G theo tham số u
- **n** là vector xác định hướng tạo sinh mặt cong

- Điều kiện: $0 \leq u \leq u_{\max}$ và $0 \leq v \leq v_{\max}$

Rulesurf: $\mathbf{P}(\mathbf{u}, \mathbf{v}) = (1 - v) \cdot \mathbf{G}(\mathbf{u}) + v \cdot \mathbf{Q}(\mathbf{u})$

- $\mathbf{G}(\mathbf{u})$ là phương trình tham số của đường cong G theo tham số u
- $\mathbf{Q}(\mathbf{u})$ là phương trình tham số của đường cong Q theo tham số u
- Điều kiện: $0 \leq u \leq 1$ và $0 \leq v \leq 1$

Revsurf: $\mathbf{P}(\mathbf{u}, \mathbf{v}) = \mathbf{rz}(\mathbf{u}) \cdot \cos(\mathbf{v} \cdot \mathbf{n}_1) + \mathbf{rz}(\mathbf{u}) \cdot \sin(\mathbf{v} \cdot \mathbf{n}_2) + \mathbf{zl}(\mathbf{u}) \cdot \mathbf{u}_3$

- $\mathbf{G}(\mathbf{u})$ là phương trình tham số của đường cong G theo tham số u
- Trục \mathbf{n}_1 vuông góc với \mathbf{n}_3 (là một trục tự cho) qua đầu mút của đường sinh $\mathbf{G}(\mathbf{u})$
- Trục \mathbf{n}_2 vuông góc với mặt phẳng tạo bởi $(\mathbf{n}_1, \mathbf{n}_3)$
- $\mathbf{zl}(\mathbf{u})$: khoảng cách từ mặt phẳng $(\mathbf{n}_1, \mathbf{n}_2)$ đến mặt cắt ngang thẳng góc \mathbf{n}_3 đi qua $\mathbf{G}(\mathbf{u})$
- $\mathbf{rz}(\mathbf{u})$: khoảng cách từ điểm tại $\mathbf{G}(\mathbf{u})$ đến trục \mathbf{n}_3
- Điều kiện: $0 \leq u \leq 1, 0 \leq v \leq 2\pi$

Edgesurf: $\mathbf{P}(\mathbf{u}, \mathbf{v}) = \mathbf{P1}(\mathbf{u}, \mathbf{v}) + \mathbf{P2}(\mathbf{u}, \mathbf{v}) - \mathbf{P3}(\mathbf{u}, \mathbf{v})$

- $\mathbf{P}(\mathbf{u}, 0), \mathbf{P}(1, \mathbf{v}), \mathbf{P}(\mathbf{u}, 1), \mathbf{P}(0, \mathbf{v})$: phương trình tham số của 4 đường cong theo tham số u, v
- $\mathbf{P1}(\mathbf{u}, \mathbf{v}) = (1 - u) \cdot \mathbf{P}(0, \mathbf{v}) + u \cdot \mathbf{P}(1, \mathbf{v})$
- $\mathbf{P2}(\mathbf{u}, \mathbf{v}) = (1 - v) \cdot \mathbf{P}(\mathbf{u}, 0) + v \cdot \mathbf{P}(\mathbf{u}, 1)$
- $\mathbf{P3}(\mathbf{u}, \mathbf{v}) = (1 - v) \cdot [(1 - u) \cdot \mathbf{P}(0, 0) + u \cdot \mathbf{P}(1, 0)] + v \cdot [(1 - u) \cdot \mathbf{P}(0, 1) + u \cdot \mathbf{P}(1, 1)]$
- Điều kiện: $0 \leq u \leq 1, 0 \leq v \leq 1$

52. Giải thích vì sao, khi chiếu từ mặt sau, nếu mặt phẳng alpha có tích vô hướng với mặt chiếu > 0 thì mặt alpha đó không nhìn thấy (không được vẽ lên canva)

➔ Khi tích vô hướng này là dương, tức là lớn hơn 0, điều đó có nghĩa là góc giữa hai véc-tơ là góc nhọn (nhỏ hơn 90 độ). Khi đó, mặt phẳng alpha và mặt chiếu

đang hướng ra cùng phía hoặc gần nhau hơn. Vì thế, khi quan sát từ phía ngược lại, mặt phẳng alpha bị che khuất bởi mặt chiếu và không nhìn thấy được.

Ngược lại, nếu tích vô hướng giữa mặt phẳng alpha và mặt chiếu là âm, tức là nhỏ hơn 0, góc giữa hai véc-tơ là góc tù (lớn hơn 90 độ). Khi đó, mặt phẳng alpha và mặt chiếu đang hướng ra hai hướng trái ngược nhau hoặc xa nhau hơn. Do đó, khi quan sát từ phía ngược lại, mặt phẳng alpha không bị che khuất bởi mặt chiếu và có thể nhìn thấy được.

53. Xác định khi nào 1 đỉnh nằm trong clipping window, khi nào nằm ngoài clipping window.

Theo Cohen-Sutherland, để một điểm nằm trong clipping window thì điểm đó phải có tung độ y thỏa $y_{\min} < y < y_{\max}$ và hoành độ x thỏa $x_{\min} < x < x_{\max}$.

Không thỏa một trong hai điều kiện trên, điểm đó sẽ nằm ngoài vùng clipping window.

54. Viết giải thuật line clipping

Region code là một dãy bit 4-bit để xác định vị trí tương ứng so với vùng “Window”

Input: Vùng “Window”, một đoạn thẳng

Bước 1: Gán “region code” cho hai đầu mút của đoạn thẳng

Bước 2: Thực hiện phép **OR** cho “region code” tìm được ở Bước 1

Bước 3:

- Nếu kết quả của bước 2 là “0000” thì kết thúc.
- Nếu không thì chuyển sang bước 4.

Bước 4: Thực hiện phép toán tử **AND** cho “region code” tìm được ở Bước 1

Bước 5:

- Nếu kết quả của bước 4 khác “0000” thì kết thúc.
- Nếu không thì chuyển sang bước 6.

Bước 6: Đi tính hằng số độ dốc đoạn thẳng là $m = (y_2 - y_1) / (x_2 - x_1)$

Bước 7:

- Nếu đường thẳng đi qua biên giới trên của “Window”
 - $x = x + (y_{wmax} - y) / m$
 - $y = y_{wmax}$
- Nếu đường thẳng đi qua biên giới dưới của “Window”
 - $x = x + (y_{wmin} - y) / m$
 - $y = y_{wmin}$
- Nếu đường thẳng đi qua biên giới trái của “Window”
 - $x = x_{wmin}$
 - $y = y + (x_{wmin} - x) * m$
- Nếu đường thẳng đi qua biên giới phải của “Window”
 - $x = x_{wmax}$
 - $y = y + (x_{wmax} - x) * m$

Bước 8: Gán lại các điểm đầu mút bằng x, y vừa tìm được và quay về bước 1

Output: Một đường thẳng đã được clip trong vùng “Window”