

Bộ ngân hàng câu hỏi

Môn Đồ họa máy tính

Mục lục

I. Mô tả.....	2
II. Trắc nghiệm.....	2
II.1. Thiết bị (41)	2
II.2. Các thuật toán tô màu	12
II.3. Các thuật toán vẽ đường thẳng và đường tròn (28).....	16
II.4. Các thuật toán cắt xén (19).....	24
II.5. Các phép biến đổi (36)	29
II.6. Các thuật toán xác định bề mặt hiện (13)	37
II.7. Đường cong và bề mặt (14).....	40
II.8. Ánh sáng (11)	43
II. 9 OpenGL	46
III. Phần tự luận	56
III.1. Các thuật toán tô màu.....	56
III.2 Các thuật toán vẽ đường thẳng/ vẽ đường tròn	61
III.3. Các thuật toán cắt xén	68
III.4. Các phép biến đổi/ đường cong và bề mặt.....	72
III.5. Các thuật toán xác định bề mặt hiện	76
III.6 Ánh sáng/ OpenGL	80
IV. Ma trận đề thi.....	83

I. Mô tả

Ngân hàng câu hỏi môn Đồ họa máy tính gồm 2 phần Trắc nghiệm và Tự luận tương ứng với hai mục II và III tiếp theo

Số lượng câu hỏi được thống kê theo bảng dưới đây

Số TT	Kho	Trắc nghiệm	Tự luận
1	Thiết bị	41	
2	Các thuật toán tô màu	21	6
3	Các thuật toán vẽ đường thẳng, đường tròn	28	7
4	Các thuật toán cắt xén	19	6
5	Các phép biến đổi	36	10
6	Đường cong và bề mặt	14	
7	Xác định bề mặt hiện	13	5
8	Ánh sáng	11	6
9	OpenGL	43	
Tổng số		226	40

Phần IV đưa ra ma trận đề thi để đánh giá cuối kỳ cho mỗi khóa học.

II. Trắc nghiệm

II.1. Thiết bị (41)

Câu 1.

H	Thiết bị đầu vào thường dùng nhất là
Đ	Bàn phím (Keyboard)
T1	Chuột (Mouse)
T2	Máy quét (Scanner)
T3	Máy in (Printer)
K	2
M	0.25

Câu 2.

H	Những phím nào cho phép người dùng nhập các hoạt động thường xuyên sử dụng trong một lần ấn phím (key stroke)?
Đ	Phím chức năng (Function keys)
T1	Phím điều khiển con trỏ (Cursor control keys)
T2	Thiết bị chỉ “Trackball” (Trackball)
T3	Phím điều khiển (Control keys)
K	2
M	0.25

Câu 3.

H	Điền vào phần bỏ trống _____ được dùng để đo độ xoay vòng quay.
Đ	Chỉ đáp án {T1}
T1	Chiết áp (Potentiometers)
T2	Máy đo điện thế (Volta meter)
T3	Tham số (Parameter)
K	2
M	0.25

Câu 4.

H	Thiết bị dùng để chỉ vị trí con trỏ trên màn hình là
Đ	Chuột (Mouse)
T1	Phím điều khiển trò chơi (Joystick)
T2	Găng tay cảm nhận (Data glove)
T3	Cả {T1} và {T2}
K	2
M	0.25

Câu 5.

H	Điền vào phần bỏ trống _____ được dùng để phát hiện chuyển động của chuột.
Đ	Cả {T1} và {T2}
T1	Cảm biến quang học (Optical sensor)
T2	Bi cuộn dưới đáy chuột (Rollers on the bottom of mouse)
T3	Cảm biến (Sensor)
K	2
M	0.25

Câu 6.

H	Trackball là
Đ	Thiết bị trỏ vị trí 2 chiều (Two- dimensional positioning device)
T1	Thiết bị trỏ vị trí 3 chiều (Three- dimensional positioning device)
T2	Thiết bị trỏ (Pointing device)
T3	Không lựa chọn nào ở trên
K	2
M	0.25

Câu 7.

H	Điền vào phần bỏ trống Space ball cung cấp _____ bậc tự do.
Đ	6 bậc
T1	10 bậc

T2	8 bậc
T3	12 bậc
K	2
M	0.25

Câu 8.

H	Loại thiết bị nào có thể xoay bằng cách xoay với ngón tay hoặc lòng bàn tay?
Đ	Trackball
T1	Space ball
T2	Chỉ {T1}
T3	Không phải {Đ}
K	2
M	0.25

Câu 9.

H	Điền vào phần bỏ trống _____ được sử dụng cho trò 3 chiều, mô hình hóa, hoạt cảnh và các ứng dụng khác.
Đ	Space ball
T1	Trackball
T2	Spac ball
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 10.

H	Chiết áp được gắn vào phần đế của phím điều khiển trò chơi để đo
Đ	Lượng chuyển động (The amount of movement)
T1	Hướng chuyển động (The direction)
T2	Vị trí (Position)
T3	Độ phân giải (Resolution)
K	2
M	0.25

Câu 11.

H	Phím điều khiển trò chơi nhạy cảm áp suất (Pressure-sensitive joysticks) còn được gọi là
Đ	Phím điều khiển đối xứng (Isometric joystick)
T1	Phím không chuyển động (Non movable stick)
T2	Phím điều khiển (Joystick)
T3	Không đáp án nào nêu trên
K	2

M	0.25
---	------

Câu 12.

H	Thiết bị nào dưới đây được xây dựng với các cảm biến dùng để phát hiện chuyển động bàn tay và ngón tay?
Đ	Găng tay cảm nhận (Data glove)
T1	Bộ số hóa (Digitizers)
T2	Phím điều khiển trò chơi (Joystick)
T3	Track ball
K	2
M	0.25

Câu 13.

H	Thiết bị phổ biến dùng để vẽ, tô màu hoặc lựa chọn vị trí tọa độ trên một đối tượng là
Đ	Bộ số hóa (Digitizers)
T1	Máy quét ảnh (Image scanner)
T2	Găng tay cảm nhận (Data glove)
T3	Bảng lựa chọn bằng tiếp xúc (Touch panels)
K	2
M	0.25

Câu 14.

H	Thiết bị nào được sử dụng làm đầu vào là tọa độ 2 chiều bằng cách kích hoạt con trỏ tay trên một bề mặt phẳng?
Đ	Cả {T1} và {T2}
T1	Graphic tablet
T2	Data tablet
T3	Chỉ {T2}
K	2
M	0.25

Câu 15.

H	Điền vào phần bỏ trống _____ được sử dụng để xác định vị trí trên data tablet.
Đ	Hoặc Signal strength hoặc coded pulse
T1	Strip microphones
T2	Signal strength
T3	Coded pulse
K	2
M	0.25

Câu 16.

H	Điền vào phần bỏ trống _____ cho phép vị trí trên màn hình được lựa chọn bằng việc tiếp xúc một ngón tay.
Đ	Bảng lựa chọn bằng tiếp xúc (Touch panels)
T1	Máy quét ảnh (Image scanner)
T2	Light pen
T3	Mouse
K	2
M	0.25

Câu 17.

H	Nhược điểm của light pen là gì?
Đ	Không thể phát hiện vị trí trong vùng màu đen
T1	Hình dáng
T2	Không thể phát hiện vị trí
T3	Đọc chính xác
K	2
M	0.25

Câu 18.

H	Điền vào phần bỏ trống _____ Được sử dụng trong trạm làm việc đồ họa như là thiết bị vào chấp nhận các câu lệnh bằng giọng nói.
Đ	Bộ nhận dạng tiếng nói (Speech recognizers)
T1	Bảng lựa chọn bằng tiếp xúc (Touch panels)
T2	Chỉ a
T3	Tất cả các lựa chọn ở trên
K	2
M	0.25

Câu 19.

H	Giọng nói được sử dụng để làm gì trong hệ thống dùng giọng nói?
Đ	Cả {T1} và {T2}
T1	Để khởi tạo các thao tác đồ họa (To initiate graphics operation)
T2	Để nhập dữ liệu (To enter data)
T3	Không a cũng không b
K	2
M	0.25

Câu 20.

H	Điền vào phần bỏ trống
---	------------------------

	Khi một câu lệnh bằng giọng nói được đưa ra, hệ thống sẽ tìm kiếm trong_____ để đối sánh mẫu tần số.
Đ	Từ điển (Dictionary)
T1	Bộ nhớ (Memory)
T2	Dữ liệu vào (Input data)
T3	Đĩa cứng (Hard disk)
K	2
M	0.25

Câu 21.

H	Thiết bị được thiết kế để tối thiểu hóa âm thanh nền là
Đ	Micro (Microphone)
T1	Bộ số hóa (Digitizers)
T2	Găng tay cảm nhận (Data glove)
T3	Phím điều khiển trò chơi (Joy stick)
K	2
M	0.25

Câu 22.

H	Chất lượng một bức ảnh có được từ thiết bị phụ thuộc vào
Đ	Tất cả các lựa chọn {T1}, {T2} và {T3}
T1	Kích cỡ của 1 điểm (Dot size)
T2	Số lượng điểm trên mỗi inch
T3	Số lượng dòng trên mỗi inch
K	2
M	0.25

Câu 23.

H	Thiết bị nào dưới đây không phải thiết bị vào?
Đ	Dòng máy in tác động (Impact printers)
T1	Trackball và space ball
T2	Găng tay cảm nhận (Data glove)
T3	
K	2
M	0.25

Câu 24.

H	Thiết bị nào dưới đây có chứa thumbwheel, trackball và mouse ball chuẩn?
Đ	Chuột Z (Z mouse)
T1	Điều khiển trò chơi (Joystick)
T2	Chuột (Mouse)

T3	Trackball
K	2
M	0.25

Câu 25.

H	Thực tế ảo, CAD, và hoạt cảnh là ứng dụng của
Đ	Chuột Z (Z mouse)
T1	Bộ số hóa (Digitizers)
T2	Bảng dữ liệu (Data tablets)
T3	Máy quét ảnh (Image scanner)
K	2
M	0.25

Câu 26.

H	Thiết bị nào cung cấp thông tin vị trí cho hệ thống đồ họa?
Đ	Cả {T1} và {T3}
T1	Thiết bị vào (Input devices)
T2	Thiết bị ra (Output devices)
T3	Thiết bị trỏ (Pointing devices)
K	2
M	0.25

Câu 27.

H	Số lượng điểm ảnh được lưu trữ trong bộ đệm khung của một hệ thống đồ họa được biết đến như là
Đ	Độ phân giải (Resolution)
T1	Độ sâu (Depth)
T2	Tốc độ làm tươi (Refresh rate)
T3	Dung lượng bộ nhớ (Capacity of memory)
K	2
M	0.25

Câu 28.

H	Trong hệ thống đồ họa, mảng các điểm ảnh trong một bức ảnh được lưu trữ trong
Đ	Bộ đệm khung (Frame buffer)
T1	Bộ nhớ (Memory)
T2	Bộ vi xử lý (Processor)
T3	Tất cả các lựa chọn {T1} và {T2}
K	2
M	0.25

Câu 29.

H	Nhiệt được cung cấp cho ống ca tốt (cathode) bằng một dòng điện đi qua dây điện được gọi là
Đ	Dây tóc (Filament)
T1	Súng điện tử (Electron gun)
T2	Dòng điện tử (Electron beam)
T3	A nôt và ca tốt (Anode and cathode)
K	2
M	0.25

Câu 30.

H	Số lượng tối đa các điểm có thể hiển thị không bị trung nhau trong ống CRT được coi như là
Đ	Độ phân giải (Resolution)
T1	Ảnh (Picture)
T2	Đại lượng không đổi (Persistence)
T3	{T1} hoặc {T2}
K	2
M	0.25

Câu 31.

H	Điền vào phần bỏ trống _____ lưu trữ thông tin ảnh như là phân bố sự thay đổi đẳng sau màn hình phốt pho.
Đ	Ống lưu trữ cảnh trực tiếp (Direct-view storage tube)
T1	Ống tia ca tốt (Cathode ray tube)
T2	Màn hình hiển thị phẳng (Flat panel displays)
T3	Thiết bị hiển thị ba chiều (3D viewing device)
K	2
M	0.25

Câu 32.

H	Thiết bị chuyển đổi từ năng lượng điện thành ánh sáng được gọi là
Đ	Bộ phát (Emitters)
T1	Màn hình tinh thể lỏng (Liquid-crystal displays)
T2	Bộ không phát (Non-emitters)
T3	Màn hình Plasma (Plasma panels)
K	2
M	0.25

Câu 33.

H	Trong hệ thống nào, phương pháp mặt nạ tạo bóng (the Shadow mask methods) thường được sử dụng
Đ	Hệ thống quét màn hình (Raster-scan system)
T1	Hệ thống quét ngẫu nhiên (Random-scan system)
T2	Chỉ {T1}
T3	d) Cả {Đ} và {T1}
K	2
M	0.25

Câu 34.

H	Quá trình xử lý số hóa một ảnh cho trước thành tập cường độ điểm ảnh lưu trữ trong bộ đệm khung được gọi là
Đ	Chuyển đổi quét (Scan conversion)
T1	Màn hình hóa (Rasterization)
T2	Mã hóa (Encoding)
T3	Hệ thống màu thực (True color system)
K	2
M	0.25

Câu 35.

H	Thiết bị hiển thị nào cho phép di chuyển xung quanh đối tượng và hiển thị đối tượng từ các hướng khác nhau.
Đ	Thiết bị hiển thị ba chiều (Three-dimensional devices)
T1	Ống lưu trữ cảnh trực tiếp (Direct view storage tubes)
T2	Thiết bị hiển thị màn hình phẳng (Flat panel display devices)
T3	Thiết bị hiển thị plasma (Plasma panel display devices)
K	2
M	0.25

Câu 36.

H	Trong màn hình LCD, tốc độ làm tươi của màn hình là
Đ	60 khung hình /s
T1	80 khung hình /s
T2	30 khung hình /s
T3	100 khung hình /s
K	2
M	0.25

Câu 37.

H	Hệ thống quét ngẫu nhiên được thiết kế cho
Đ	Ứng dụng vẽ đường thẳng (Line-drawing applications)
T1	Màn hình đổ bóng thực (Realistic shaded screen)

T2	Hiệu ứng sương mù (Fog effect)
T3	Chỉ {T2}
K	2
M	0.25

Câu 38.

H	38. Điền vào phần bỏ trống Thiết bị ra chính trong hệ thống đồ họa là _____
Đ	Màn hình hiển thị video (Video monitor)
T1	Máy quét
T2	Không {Đ} cũng không {T1}
T3	Máy in
K	2
M	0.25

Câu 39.

H	Trên một hệ thống đen trắng với một bit trên một điểm ảnh (one bit per pixel), bộ đệm khung thường được gọi là
Đ	Bitmap
T1	Pix map
T2	Multi map
T3	Tất cả các lựa chọn {Đ}, {T1} và {T2}
K	2
M	0.25

Câu 40.

H	“Aspect ratio” có nghĩa là
Đ	Cả {T2} và {T3}
T1	Số lượng điểm ảnh (Number of pixels)
T2	Tỉ lệ điểm ảnh chiều cao so với chiều rộng (Ratio of vertical points to horizontal points)
T3	Tỉ lệ điểm ảnh chiều rộng so với chiều cao (Ratio of horizontal points to vertical points)
K	2
M	0.25

Câu 41.

H	Giao diện đồ họa cơ bản là
Đ	Cả {T1} và {T2}
T1	Thiết bị trỏ (Pointing)
T2	Thiết bị định vị (Positioning)
T3	Không lựa chọn nào ở trên

K	2
M	0.25

II.2. Các thuật toán tô màu

Câu 1.

H	Các lựa chọn màu được mã hóa số với các giá trị sau.
Đ	Miền bao gồm 0 và các số nguyên dương
T1	Miền từ 0 tới 1
T2	Miền từ 0 tới -0
T3	Chỉ {T2}
K	2
M	0.25

Câu 2.

H	Trong hệ thống màn hình có màu, số lượng màu sẵn có phụ thuộc vào
Đ	Dung lượng cung cấp để lưu trữ 1 điểm ảnh trong bộ đệm khung
T1	Màu trong bộ đệm khung
T2	Màu RGB
T3	Không a cũng không b
K	2
M	0.25

Câu 3.

H	Màu có mã “000” là màu
Đ	Đen
T1	Trắng
T2	Xanh dương
T3	Xanh lá
K	2
M	0.25

Câu 4.

H	Thông tin màu được lưu trữ trong
Đ	Bộ đệm khung
T1	Bộ nhớ chính
T2	Bộ nhớ phụ
T3	Card đồ họa
K	2
M	0.25

Câu 5

H	Bất cứ khi nào một mã màu cụ thể được mô tả rõ trong một ứng dụng, giá trị nhị phân tương ứng được lưu trữ trong?
Đ	Trực tiếp trong bộ đệm khung (Directly in frame buffer)
T1	Bảng tra màu (Color look-up table)
T2	{Đ} hoặc {T1}
T3	Bảng tra video (Video lookup table)
K	2
M	0.25

Câu 6

H	Miền giá trị mô tả các mức xám
Đ	Miền giá trị từ 0 tới 1
T1	Miền giá trị từ -1 tới 1
T2	Miền giá trị từ 0 tới -1
T3	Bất cứ lựa chọn nào ở trên
K	2
M	0.25

Câu 7

H	Với 3 bit cho một điểm ảnh, chúng ta có thể biểu diễn 8 mức xám. Nếu chúng ta sử dụng 8 bit cho một điểm ảnh thì có thể biểu diễn được bao nhiêu mức xám?
Đ	256 mức xám
T1	18 mức xám
T2	128 mức xám
T3	Không màu
K	2
M	0.25

Câu 8

H	Với cường độ hiển thị tương ứng với màu cho trước, chỉ số ci được tính như sau
Đ	$\text{Intensity} = 0.5[\min(r, g, b) + \max(r, g, b)]$
T1	$\text{Intensity} = 0.5[\max(r, g, b) + \max(r, g, b)]$
T2	$\text{Intensity} = 0.5[\min(r, g, b) + \min(r, g, b)]$
T3	$\text{Intensity} = 0.5[\max(r, g, b) - \max(r, g, b)]$
K	2
M	0.25

Câu 9

H	Một người sử dụng có thể thiết lập đầu vào bảng màu trong chương trình ứng dụng PHIGS với hàm
Đ	setColourRepresentation (ws, ci, colorptr)
T1	setColorRepresentation (ws, ci, colorptr)

T2	setColour (ws, ci, colorptr)
T3	setColourRepresentation ()
K	2
M	0.25

Câu 10

H	Nếu bất cứ giá trị cường độ vào gần 0.33 có thể được lưu trữ thành giá trị nhị phân 1 trong bộ đệm khung thì nó biểu
Đ	Màu xám tối
T1	Màu xanh lá tối
T2	Màu xám sáng
T3	Trắng hoặc đen
K	2
M	0.25

Câu 11

H	Một kiểu tô màu cơ bản trong chương trình PHIGS là hàm
Đ	setInteriorStyle (fs)
T1	setStyle (fs)
T2	SetfillStyle (fs)
T3	setInteriorStyleIndex (fs)
K	2
M	0.25

Câu 12

H	Loại này không phải kiểu tô màu cơ bản?
Đ	Tối (Dark)
T1	Rỗng (Hollow)
T2	Màu đặc (solid color)
T3	Mẫu (Pattern)
K	2
M	0.25

Câu 13

H	13. Quá trình tô màu một vùng với mẫu hình chữ nhật được gọi là
Đ	Tiling
T1	Linear fill
T2	Tint-fill
T3	Soft-fill
K	2
M	0.25

Câu 14

H	Thuật toán tô màu lại một vùng mà trước đó đã được tô bằng cách trộn màu “foreground color F” và ”background color B” khi $F \neq B$ là.
Đ	Linear soft-fill
T1	Tint fill
T2	Flood fill
T3	Boundary fill
K	2
M	0.25

Câu 15

H	Màu tô kết hợp với màu nền “ là
Đ	Cả {T1} và {T2}
T1	Soft fill
T2	Tint fill
T3	Không lựa chọn nào
K	2
M	0.25

Câu 16

H	Thủ tục tô màu vân “Hatch fill” được gọi để vẽ
Đ	Hoặc {T1} hoặc {T2}
T1	Single hatching
T2	Cross hatching
T3	Chỉ {T2}
K	2
M	0.25

Câu 17

H	Khi hai màu nền (background color) B1 và B2 được trộn với màu ”foreground color F”, kết quả được màu P được tính như sau
Đ	Only {T1}
T1	$P = t_0 * F + t_1 * B_1 + (1 - t_0 - t_1) B_2$
T2	$P = t_0 * F - t_1 * B_1 + (1 - t_0 - t_1) B_2$
T3	$P = t_0 * F + t_1 * B_1 + (1 + t_0 + t_1) B_2$
K	2
M	0.25

Câu 18

H	Phép toán được dùng để kết hợp mẫu tô (fill pattern) với mẫu nền (background pattern) là
---	--

Đ	Tất cả các lựa chọn {T1}, {T2} và {T3}
T1	a) AND
T2	b) OR
T3	c) X-OR
K	2
M	0.25

Câu 19

H	Vùng rỗng (Hollow areas) được hiển thị chỉ dùng
Đ	Viền biên (Boundary outline)
T1	Lộ trình vẽ đường thẳng (Line-drawing routine)
T2	Mẫu vân sọc (Hatched patterns)
T3	Chuỗi các đường thẳng khép kín (Closed poly line)
K	2
M	0.25

Câu 20

H	Các lựa chọn cho tô màu một vùng được định nghĩa bao gồm một lựa chọn giữa _____
Đ	Cả {T1} và {T2}
T1	Màu đặc hoặc mẫu (Solid color or a pattern fill)
T2	Các lựa chọn cho các màu và mẫu cụ thể (Choices for particular colors and pattern)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 21

H	Các dòng quét được sử dụng để quét từ
Đ	Đỉnh tới đáy (Top to bottom)
T1	Đáy tới đỉnh (Bottom to top)
T2	Cả {Đ} và {T1}
T3	Không lựa chọn nào
K	2
M	0.25

II.3. Các thuật toán vẽ đường thẳng và đường tròn (28)

Câu 1

H	Phương trình đường thẳng dưới dạng hệ số góc (Cartesian slope-intercept equation) có dạng như sau
---	---

Đ	$y = m.x + b$
T1	$y = b.x + m$
T2	$y = x.x + m$
T3	$y = b + m.m$
K	2
M	0.25

Câu 2

H	Đối với các đường thẳng có hệ số góc m với $ m < 1$, khi áp dụng thuật toán đơn giản để vẽ đường thẳng thì
Đ	Tăng x lên 1 đơn vị rồi tính y
T1	Tăng y lên 1 đơn vị rồi tính x
T2	Cả $\{Đ\}$ và $\{T1\}$
T3	Không lựa chọn nào ở trên
K	2
M	0.25

Câu 3

H	Trong hệ thống màn hình hóa, các đường thẳng được vẽ dưới dạng
Đ	Điểm ảnh
T1	Đường thẳng
T2	Chấm
T3	Không lựa chọn nào ở trên
K	2
M	0.25

Câu 4

H	Tên đầy đủ của DDA trong thuật toán DDA là
Đ	Digital differential analyzer
T1	Digital difference analyzer
T2	Direct differential analyzer
T3	Data differential analyzer
K	2
M	0.25

Câu 5

H	Thuật toán nào là phương pháp nhanh nhất tìm vị trí của điểm ảnh khi vẽ đường thẳng trong số các thuật toán dưới đây?
Đ	Thuật toán vẽ đường thẳng Bresenham

T1	Thuật toán vẽ đường thẳng dùng đường thẳng hệ số góc đơn giản
T2	Thuật toán điểm giữa vẽ đường thẳng
T3	Thuật toán DDA vẽ đường thẳng
K	2
M	0.25

Câu 6

H	Một thuật toán mảnh hóa vẽ đường thẳng chính xác và hiệu quả là
Đ	Thuật toán vẽ đường thẳng Bresenham
T1	Thuật toán vẽ đường thẳng dùng đường thẳng hệ số góc đơn giản
T2	Thuật toán điểm giữa vẽ đường thẳng
T3	Thuật toán DDA vẽ đường thẳng
K	2
M	0.25

Câu 7

H	Trong thuật toán Bresendham vẽ đường thẳng, Nếu $d1 < d2$ thì tham số quyết định P_k là_____
Đ	Âm
T1	Dương
T2	Tương đương
T3	{Đ} hoặc {T1}
K	2
M	0.25

Câu 8

H	Thuật toán vẽ đường thẳng nào là tốt nhất trong việc cân bằng tải xử lý giữa các bộ xử lý?
Đ	Thuật toán song song vẽ đường thẳng
T1	Thuật toán vẽ đường thẳng Bresenham
T2	Thuật toán điểm giữa vẽ đường thẳng
T3	Thuật toán DDA vẽ đường thẳng
K	2
M	0.25

Câu 9

H	Thuật toán sử dụng nhiều bộ xử lý để tính toán các vị trí của điểm ảnh là
Đ	Thuật toán song song vẽ đường thẳng

T1	Thuật toán vẽ đường thẳng Bresenham
T2	Thuật toán điểm giữa vẽ đường thẳng
T3	Thuật toán DDA vẽ đường thẳng
K	2
M	0.25

Câu 10

H	Tham chiếu tọa độ trong hàm vẽ chuỗi đường thẳng được chỉ ra như
Đ	Giá trị tọa độ tuyệt đối (Absolute coordinate values)
T1	Giá trị tọa độ liên quan (Relative coordinate values)
T2	Vị trí hiện tại (Current position)
T3	Giá trị tọa độ thực (Real coordinate values)
K	2
M	0.25

Câu 11

H	Để áp dụng phương pháp điểm giữa vẽ đường tròn, chúng ta định nghĩa
Đ	$\text{circle}(x, y) = x^2 + y^2 - r^2$
T1	$\text{circle}(x, y) = x + y^2 - r^2$
T2	$\text{circle}(x, y) = x^2 + y - r$
T3	$\text{circle}(x, y) = x^2 + y^2 - z^2$
K	2
M	0.25

Câu 12

H	Điền vào phần bỏ trống _____ được định nghĩa là tập các điểm sao tổng khoảng cách từ điểm đó tới 2 điểm cố định cho trước không đổi.
Đ	Elip
T1	Đường thẳng
T2	Đường tròn
T3	Tam giác
K	2
M	0.25

Câu 13

H	Nếu đường biên được mô tả với một màu, thì thuật toán xuất phát từ một điểm xử lý từng điểm ảnh cho đến khi gặp được màu biên được gọi là
---	---

Đ	Thuật toán tô màu loang (Flood-fill algorithm)
T1	Thuật toán tô màu dòng quét (Scan-line fill algorithm)
T2	Thuật toán tô biên (Boundary-fill algorithm)
T3	Thuật toán đường cong song song (Parallel curve algorithm)
K	2
M	0.25

Câu 14

H	Thuật toán tô phủ loang dùng quan hệ láng giềng nào
Đ	c) Cả {T1} và {T2}
T1	4 láng giềng
T2	8 láng giềng
T3	d) Chỉ {T1}
K	2
M	0.25

Câu 15

H	Thuộc tính cơ bản của đoạn thẳng là
Đ	Tất cả lựa chọn {T1}, {T2} và {T3}
T1	Kiểu
T2	Độ rộng
T3	Màu
K	2
M	0.25

Câu 16

H	Một đường thẳng tô nét đứt (dashed line) được hiển thị bằng cách sinh ra _____.
Đ	Inter dash spacing
T1	Very short dashes
T2	Cả {T1} và {T2}
T3	d) {T1} hoặc {T2}
K	2
M	0.25

H	Một đường thẳng tô dưới dạng điểm (dotted line) có thể được hiển thị bằng cách sinh ra
Đ	Very short dashes với khoảng trắng lớn hơn hoặc bằng cỡ của dash
T1	Very short dashes với khoảng trắng nhỏ hơn hoặc bằng cỡ của dash

T2	Very long dashes với khoảng trắng lớn hơn hoặc bằng cỡ của dash
T3	Dots
K	2
M	0.25

Câu 17

H	Lựa chọn nào dưới đây không phải kiểu đường thẳng (line-type)?
Đ	Chỉ {T2}
T1	Dashed line
T2	Dark line
T3	Dotted line
K	2
M	0.25

Câu 18

H	Trong một ứng dụng, để thiết lập thuộc tính kiểu đường thẳng (line-type) chúng ta dùng câu lệnh
Đ	setLinetype(lt)
T1	SetLinetype(lt)
T2	SETLINETYPE(lt)
T3	SETLINE()
K	2
M	0.25

Câu 19

H	Thuật toán hiển thị thuộc tính kiểu đường thẳng (line-type) bằng cách vẽ từng nhíp điểm ảnh (pixel span) là
Đ	Raster line algorithm
T1	Raster scan algorithm
T2	Random line algorithm
T3	Random scan algorithm
K	2
M	0.25

Câu 20

H	Mặt nạ điểm ảnh (Pixel mask) có nghĩa là
Đ	Chuỗi chứa 0 và 1
T1	Chuỗi chỉ chứa 1
T2	Chuỗi chỉ chứa 0
T3	

K	2
M	0.25

Câu 21

H	Một “heavy line” trên màn hình video có thể hiển thị như
Đ	Adjacent parallel lines
T1	Adjacent perpendicular lines
T2	Cả {Đ} và {T1}
T3	Hoặc {Đ} hoặc {T1}
K	2
M	0.25

Câu 22

H	Tham số cho “setLineWidthScaleFactor (lw) “ được mô tả là?
Đ	Độ rộng của đường thẳng đang vẽ (Relative width of the line)
T1	Độ rộng chuẩn (Standard width)
T2	Độ dày của đường thẳng đang vẽ (Thickness of the line)
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 23

H	Chúng ta có thể chỉnh hình dáng của các điểm cuối (line ends) của đường thẳng có bề ngoài tốt hơn sử dụng
Đ	Line caps
T1	Line spacing
T2	More dots
T3	Round cap
K	2
M	0.25

Câu 24

H	Độ dày đường thẳng (Thick line) được vẽ với
Đ	Tất cả các lựa chọn trên
T1	Butt caps
T2	Round caps
T3	Projecting square caps
K	2
M	0.25

Câu 25

H	Chúng ta thiết lập giá trị màu đường thẳng (line-color value) trong PHIGS bằng hàm
Đ	setPolylineColorIndex (lc)
T1	setline Color()
T2	SETPOLYLINECOLORINDEX (lc)
T3	d) Chỉ {T1}
K	2
M	0.25

Câu 26

H	Điền vào phần bỏ trống Nếu góc giữa hai đoạn thẳng nối nhau là rất nhỏ thì _____ có thể sinh ra sự tăng đột biến làm biến dạng bề ngoài của polyline.
Đ	Miter join
T1	Round join
T2	Bevel join
T3	Không lựa chọn nào
K	2
M	0.25

Câu 27

H	Một đường thẳng được vẽ với màu nền thì
Đ	Invisible
T1	Visible
T2	Visible or Invisible
T3	Một đường thẳng được vẽ với màu nền thì
K	2
M	0.25

Câu 28

H	Giá trị của tham số t nằm giữa
Đ	1 và 10
T1	1 và 2
T2	0 và 1
T3	0 và 3
K	2
M	0.25

II.4. Các thuật toán cắt xén (19)

Câu 1

H	Quá trình ánh xạ từ cửa sổ thế giới từ hệ tọa độ thế giới sang viewport được gọi là
Đ	Transformation viewing
T1	Viewport
T2	Clipping window
T3	Screen coordinate system
K	2
M	0.25

Câu 2

H	Quá trình trích chọn một phần của cơ sở dữ liệu hoặc một ảnh bên trong/ bên ngoài của một vùng được gọi là
---	--

Đ	Clipping
T1	Transformation
T2	Projection
T3	Mapping
K	2
M	0.25

Câu 3

H	Phần hình chữ nhật của một cửa sổ giao diện nơi mà bức ảnh sẽ xuất hiện được gọi là
Đ	View port
T1	Transformation viewing
T2	Clipping window
T3	Screen coordinate system
K	2
M	0.25

Câu 4

H	Hệ tọa độ của cửa sổ nơi đối tượng thật xuất hiện được gọi là
Đ	World coordinates
T1	Screen coordinates
T2	Device coordinates
T3	Cartesian coordinates
K	2
M	0.25

Câu 5

H	Hệ tọa độ của viewport được gọi là
Đ	Screen coordinates
T1	World coordinates
T2	Polar coordinates
T3	Cartesian coordinates
K	2
M	0.25

Câu 6

H	Vùng chứa đối tượng sau khi bị cắt xén được gọi là
Đ	Clip window
T1	Boundary
T2	Enclosing rectangle
T3	Clip square
K	2

M	0.25
---	------

Câu 7

H	Điền vào phần bỏ trống để xác định vùng ảnh nằm ngoài cửa sổ cắt xén (clip window)
Đ	Exterior clipping
T1	Interior clipping
T2	Extraction
T3	Không lựa chọn nào ở trên
K	2
M	0.25

Câu 8

H	Xác định thuật toán dùng để cắt xén đoạn thẳng
Đ	Tất cả các lựa chọn {T1}, {T2} và {T3}
T1	Cohen- Sutherland algorithm
T2	Liang-Barsky clipping
T3	Nicholl-Lee-Nicholl clipping
K	2
M	0.25

Câu 9

H	Điền vào phần bỏ trống Mã dùng để mã hóa vùng cửa sổ cắt xén là
Đ	0000
T1	1111
T2	1000
T3	0001
K	2
M	0.25

Câu 10

H	Theo thuật toán Cohen-Sutherland, một đoạn thẳng hoàn toàn nằm bên ngoài cửa sổ cắt xén nếu
Đ	Mã của hai đầu mút đoạn thẳng có cùng bit '1' tại cùng vị trí.
T1	Mã của hai đầu mút đoạn thẳng có giá trị khác 0
T2	Nếu bit L và R khác 0.
T3	Mã của hai đầu mút đoạn thẳng có cùng bit '0' tại cùng vị trí.
K	2
M	0.25

Câu 11

H	Điền vào phần bỏ trống
---	------------------------

	Mã của một điểm là 1001. Điểm đó nằm ở vùng so với cửa sổ cắt xén.
Đ	Top left
T1	Top right
T2	Bottom left
T3	Bottom right
K	2
M	0.25

Câu 12

H	Kết quả của phép toán AND với mã của các đầu mút với giá trị khác 0. Câu nào sau đây đúng?
Đ	Đoạn thẳng nằm hoàn toàn bên ngoài cửa sổ cắt
T1	Đoạn thẳng nằm hoàn toàn bên trong cửa sổ cắt
T2	Đoạn thẳng có một phần nằm bên trong cửa sổ cắt
T3	Đoạn thẳng đã được cắt
K	2
M	0.25

Câu 13

H	Điền vào phần bỏ trống Bit bên trái (L bit) của mã của điểm (X,Y) là '1' nếu
Đ	$X < X_{WMIN}$
T1	$X > X_{WMIN}$
T2	$X < X_{WMAX}$
T3	$X > X_{WMAX}$
K	2
M	0.25

Câu 14

H	Bit bên phải (R bit) của mã của điểm (X,Y) là '1' nếu.....
Đ	$X > X_{WMAX}$
T1	$X > X_{WMIN}$
T2	$X < X_{WMIN}$
T3	$X < X_{WMAX}$
K	2
M	0.25

Câu 15

H	Điền vào phần bỏ trống Bit trên đỉnh (T bit) của mã của điểm (X,Y) là '1' nếu
Đ	$Y > Y_{WMAX}$
T1	$Y > Y_{WMIN}$

T2	$Y < Y_{WMIN}$
T3	$Y < Y_{WMAX}$
K	2
M	0.25

Câu 16

H	Bit dưới đáy (B bit) của mã của điểm (X,Y) là '1' nếu
Đ	$Y < Y_{WMIN}$
T1	$Y > Y_{WMIN}$
T2	$Y < Y_{WMAX}$
T3	$Y > Y_{WMAX}$
K	2
M	0.25

Câu 17

H	Điền vào phần bỏ trống Thuật toán chia không gian 2 chiều thành 9 vùng, trong đó vùng ở giữa (viewport) chứa phần hiển thị của đối tượng
Đ	Cohen-Sutherland
T1	Liang Barsky
T2	Sutherland Hodegeman
T3	N-L-N
K	2
M	0.25

Câu 18

H	Phép toán được sử dụng để kiểm tra đoạn thẳng có bị cắt xén hay không.....
Đ	logical AND
T1	logical XOR
T2	logical OR
T3	Cả {T1} và {T2}
K	2
M	0.25

H	Thuật toán Sutherland Hodgeman hoạt động tốt trong trường hợp cắt xén.....
Đ	Convex polygon
T1	Concave polygon
T2	Smooth curves
T3	Line segment
K	2
M	0.25

Câu 19

H	Điền vào phần bỏ trống Một phép biến đổi làm xiên (slants) hình dáng của đối tượng được gọi là
Đ	Shear
T1	Reflection
T2	Distortion
T3	Scaling
K	2
M	0.25

II.5. Các phép biến đổi (36)

Câu 1

H	Phép biến đổi tịnh tiến được áp dụng cho một đối tượng (object) bằng cách
Đ	Đặt lại vị trí của đối tượng dọc theo một đường thẳng
T1	Đặt lại vị trí của đối tượng dọc theo một cung tròn
T2	Chỉ {T1}
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 2

H	Chúng ta tịnh tiến một điểm 2 chiều bằng cách cộng thêm
Đ	Khoảng cách dịch chuyển
T1	Độ lệch theo X
T2	X và Y
T3	Độ lệch theo Y
K	2
M	0.25

Câu 3

H	Các khoảng cách dịch chuyển (dx, dy) được gọi là
Đ	Cả {T1} và {T2}
T1	Véc tơ dịch chuyển
T2	Véc tơ tịnh tiến
T3	Không phải lựa chọn {T1} cũng không phải lựa chọn {T2}
K	2
M	0.25

Câu 4

H	Trong tịnh tiến 2 chiều, một điểm (x, y) có thể dịch chuyển sang vị trí mới (x', y') bằng cách sử dụng phương trình
Đ	$x'=x+dx$ and $y'=y+dy$
T1	$x'=x+dx$ and $y'=y+dx$
T2	$X'=x+dy$ and $Y'=y+dx$
T3	$X'=x-dx$ and $y'=y-dy$
K	2
M	0.25

Câu 5

H	Phương trình của phép biến đổi tịnh tiến khi điểm biểu diễn dưới dạng tọa độ Đécác là
Đ	$P'=P+T$
T1	$P'=P-T$
T2	$P'=P*T$
T3	$P'=p$
K	2
M	0.25

Câu 6

H	Điền vào phần bỏ trống _____ là một phép biến đổi không biến đổi hình dạng (rigid body transformation) chỉ thay đổi vị trí đối tượng mà không làm biến dạng nó.
Đ	d) {T1} và {T3}
T1	Quay
T2	Co dãn
T3	Tịnh tiến
K	2
M	0.25

Câu 7

H	Một đoạn thẳng được tịnh tiến bằng cách áp dụng phương trình biến đổi
Đ	$P'=P+T$
T1	Dx và Dy
T2	$P'=P+P$
T3	Chỉ {T2}
K	2
M	0.25

Câu 8

H	Điền vào phần bỏ trống
---	------------------------

	Các đa giác được dịch chuyển bằng cách công thêm _____ vào vị trí tọa độ của mỗi đỉnh và thiết lập thuộc tính hiện tại.
Đ	Véc tơ tịnh tiến
T1	Đường đi một đường thẳng
T2	Độ lệch
T3	{T1} và {T3}
K	2
M	0.25

Câu 9

H	Để thay đổi vị trí của một đường tròn hay elip chúng ta tịnh tiến
Đ	Tọa độ tâm và vẽ lại hình ở vị trí mới
T1	Tọa độ tâm
T2	Các tọa độ biên
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 10

H	Các phép biến đổi hình học afin bao gồm
Đ	Tất cả các lựa chọn {T1}, {T2} và {T3}
T1	Tịnh tiến
T2	Quay
T3	Co dãn
K	2
M	0.25

Câu 11

H	Một phép quay hai chiều được áp dụng cho một đối tượng bằng cách
Đ	Đặt lại vị trí của đối tượng dọc theo một cung tròn
T1	Đặt lại vị trí của đối tượng dọc theo một đường thẳng
T2	Chỉ {T1}
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 12

H	Để sinh ra một phép quay, chúng ta phải mô tả
Đ	Góc quay Θ
T1	Khoảng cách dx và dy
T2	Khoảng cách quay

T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 13

H	Các giá trị dương trong phép quay của góc quay Θ định nghĩa
Đ	Góc quay ngược chiều kim đồng hồ xung quanh tâm quay
T1	Góc quay ngược chiều kim đồng hồ xung quanh điểm cần quay
T2	Tịnh tiến ngược chiều kim đồng hồ quanh tâm quay
T3	Hướng âm
K	2
M	0.25

Câu 14

H	Trục quay vuông góc với mặt phẳng xy và đi qua một điểm chốt được gọi là
Đ	Phép quay
T1	Phép tịnh tiến
T2	Phép có dẫn
T3	Phép kéo
K	2
M	0.25

Câu 15

H	Tọa độ Đêcac của một điểm trong hệ tọa độ cực là
Đ	$X'=r \cos (\Phi +\Theta)$ and $Y'=r \sin (\Phi +\Theta)$
T1	$X'=r \cos (\Phi +\Theta)$ and $Y'=r \cos (\Phi +\Theta)$
T2	$X'=r \cos (\Phi -\Theta)$ and $Y'=r \cos (\Phi -\Theta)$
T3	$X'=r \cos (\Phi +\Theta)$ and $Y'=r \sin (\Phi -\Theta)$
K	2
M	0.25

Câu 16

H	Phương trình phép quay 2 chiều là
Đ	$P'=R * P$
T1	$P'=P+T$
T2	$P'=P * P$
T3	$P'=R+P$
K	2
M	0.25

Câu 17

H	Điền vào phần bỏ trống
---	------------------------

	_____ là phép biến đổi di chuyển đối tượng theo quỹ đạo tròn
Đ	Quay
T1	Tĩnh tiến
T2	Co dãn
T3	Kéo
K	2
M	0.25

Câu 18

H	Một elip có thể quay quanh tâm của nó bằng cách quay
Đ	Trục lớn, trục nhỏ
T1	Các điểm đầu mút
T2	Chỉ {T1}
T3	Không lựa chọn nào ở trên
K	2
M	0.25

Câu 19

H	Chỉ ra phép biến đổi được sử dụng để thay đổi kích cỡ của đối tượng
Đ	Co dãn
T1	Quay
T2	Tĩnh tiến
T3	Phản xạ
K	2
M	0.25

Câu 20

H	Phương trình của phép co dãn 2 chiều
Đ	$P' = S * P$
T1	$P' = P + T$
T2	$P' = P * R$
T3	$P' = R + S$
K	2
M	0.25

Câu 21

H	Co dãn một hình đa giác bằng cách tính
Đ	Nhân hệ số co dãn (x, y) với từng đỉnh
T1	Chia hệ số co dãn (x,y) cho từng đỉnh
T2	Các tọa độ tâm
T3	Các tọa độ đỉnh

K	2
M	0.25

Câu 22

H	Nếu các hệ số co dãn s_x và $s_y < 1$ thì
Đ	Phép biến đổi giảm kích cỡ đối tượng
T1	Phép biến đổi tăng kích cỡ đối tượng
T2	Phép biến đổi làm xiên đối tượng
T3	Không đối tượng nào
K	2
M	0.25

Câu 23

H	Nếu giá trị của các hệ số co dãn s_x và s_y bằng nhau thì
Đ	Phép biến đổi là phép co dãn đồng nhất
T1	Phép biến đổi là phép quay đồng nhất
T2	Phép co dãn không thể thực hiện được
T3	Chỉ {T1}
K	2
M	0.25

Câu 24

H	Các đối tượng được biến đổi theo phương trình $P'=S*P$ có thể là
Đ	Cả {T1} và {T2}
T1	Bị co dãn
T2	Bị xác định lại vị trí
T3	Không lựa chọn nào
K	2
M	0.25

Câu 25

H	Chúng ta kiểm soát vị trí của đối tượng bị co dãn bằng cách chọn một điểm được biết đến là
Đ	Điểm cố định (Fixed point)
T1	Điểm xoay (Pivot point)
T2	Co dãn không đồng nhất
T3	Co dãn đồng nhất
K	2
M	0.25

Câu 26

H	Nếu các giá trị co dãn là $s_x=1$ và $s_y=1$ thì
Đ	Đối tượng không thay đổi kích cỡ

T1	Đối tượng bị giảm kích cỡ
T2	Biến dạng bức ảnh
T3	Đối tượng bị tăng kích cỡ
K	2
M	0.25

Câu 27

H	Các đa giác bị co dãn bởi áp dụng phương trình biến đổi sau.
Đ	$X'=x * S_x + X_f(1-S_x)$ and $Y'=y * S_y + Y_f(1-S_y)$
T1	$X'=x * S_x + X_f(1+S_x)$ and $Y'=y * S_y + Y_f(1+S_y)$
T2	$X'=x * S_x + X_f(1-S_x)$ and $Y'=y * S_y - Y_f(1-S_y)$
T3	$X'=x * S_x * X_f(1-S_x)$ and $Y'=y * S_y * Y_f(1-S_y)$
K	2
M	0.25

Câu 28

H	Biểu diễn dạng ma trận của phép tịnh tiến trong hệ tọa độ đồng là
Đ	$P'=T*P$
T1	$P'=T+P$
T2	$P'=S*P$
T3	$P'=R*P$
K	2
M	0.25

Câu 29

H	Nếu giá trị của các hệ số co dãn s_x và s_y không bằng nhau thì
Đ	Phép biến đổi là phép co dãn không đồng nhất
T1	Phép biến đổi là phép co dãn đồng nhất
T2	Phép co dãn không thể thực hiện được
T3	Không phải lựa chọn {Đ} cũng không phải lựa chọn {T1}
K	2
M	0.25

Câu 30

H	Biểu diễn ma trận cho biến đổi co dãn trong hệ tọa độ đồng nhất là
Đ	$P'=S*P$
T1	$P'=R*P$
T2	$P'=dx+dy$
T3	$P'=S*S$
K	2
M	0.25

Câu 31

H	Biểu diễn ma trận cho phép quay trong hệ tọa độ đồng nhất là
Đ	$P' = R * P$
T1	$P' = T + P$
T2	$P' = S * P$
T3	$P' = dx + dy$
K	2
M	0.25

Câu 32

H	Việc sử dụng của hệ tọa độ đồng nhất và biểu diễn ma trận là để
Đ	Biểu diễn các phép biến đổi theo một cách đồng nhất
T1	Thực diễn co dãn
T2	Thực hiện phép quay
T3	Thực hiện phép kéo
K	2
M	0.25

Câu 33

H	Nếu một điểm được biểu diễn trong hệ tọa độ đồng nhất thì cặp (x, y) trong hệ tọa độ Decac sẽ được biểu diễn như sau
Đ	(x', y', w)
T1	(x', y', z')
T2	(x, y, z)
T3	(x', y', w)
K	2
M	0.25

Câu 34

H	Trong phép biến đổi hai chiều giá trị thành phần tọa độ thứ 3 $w = ?$
Đ	1
T1	0
T2	-1
T3	Bất cứ giá trị nào
K	2
M	0.25

Câu 35

H	Chúng ta có thể kết hợp các phép biến đổi 2 chiều thành một phép biến đổi dùng biểu diễn các điểm trong hệ trục tọa độ đồng nhất bằng cách mở rộng biểu diễn ma trận
Đ	ma trận 2 x 2 thành ma trận 3x3
T1	ma trận 2 x 2 thành ma trận 4x4

T2	ma trận 3 x 3 thành ma trận 2x 2
T3	Chỉ {T1}
K	2
M	0.25

Câu 36

H	Biểu diễn tọa độ trong hệ trục tọa độ đồng nhất có thể viết như sau
Đ	b) (h.x, h.y, h)
T1	a) (h.x, h.y, h.z)
T2	c) (x, y, h.z)
T3	d) (x,y,z)
K	2
M	0.25

II.6. Các thuật toán xác định bề mặt hiện (13)

Câu 1

H	Các bề mặt bị khóa hoặc ẩn trong khung cảnh 3 chiều được gọi là
Đ	Các bề mặt ẩn (Hidden surface)
T1	Bộ đệm khung (Frame buffer)
T2	Cây tứ phân (Quad tree)
T3	Bề mặt bị mất (Lost surface)
K	2
M	0.25

Câu 2

H	Nhiệm vụ của bài toán loại bỏ bề mặt ẩn là
Đ	Cả {T1} và {T2}
T1	Loại bỏ bề mặt ẩn (Removal of hidden surface)
T2	Nhận diện các bề mặt ẩn (Identification of hidden surface)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 3

H	Tại sao chúng ta cần phải loại bỏ bề mặt ẩn
Đ	Cả {T1} và {T2}
T1	Để hiển thị cảnh thực
T2	Để xác định bề mặt hiện gần nhất
T3	Không lựa chọn nào
K	2
M	0.25

Câu 4

H	Thuật toán của bài toán loại bỏ bề mặt ẩn là
Đ	Cả {T1} và {T2}
T1	Chính xác theo đối tượng (Object-space method)
T2	Chính xác theo ảnh (Image-space method)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 5

H	Phương pháp dựa trên nguyên lý so sánh các đối tượng và các phần của đối tượng với nhau để tìm ra thành phần ẩn và thành phần hiển thị được gọi là
Đ	Chính xác theo đối tượng (Object-space method)
T1	Chính xác theo ảnh (Image-space method)
T2	Chính xác theo bề mặt (Surface-space method)
T3	Cả {Đ} và {T1}
K	2
M	0.25

Câu 6

H	Phương pháp dựa trên nguyên lý kiểm tra điểm hiển thị với mỗi vị trí điểm ảnh trên mặt phẳng chiếu được gọi là
Đ	Chính xác theo ảnh (Image-space method)
T1	Chính xác theo đối tượng (Object-space method)
T2	Cả {Đ} và {T1}
T3	Không lựa chọn nào
K	2
M	0.25

Câu 7

H	Các kiểu của thuật toán loại bỏ bề mặt ẩn
Đ	Tất cả lựa chọn {T1}, {T2} và {T3}
T1	So sánh độ sâu, Z-buffer, loại bỏ mặt quay vào trong (Depth comparison, Z-buffer, back-face removal)
T2	Thuật toán dòng quét, thuật toán ưu tiên (Scan line algorithm, priority algorithm)
T3	Phân hoạch nhị phân, chia nhỏ vùng (BSP method, area subdivision method)
K	2
M	0.25

Câu 8

H	Thuật toán bề mặt nào dựa trên khía cạnh độ sâu?
---	--

Đ	Z-buffer
T1	BSP
T2	Chia nhỏ vùng
T3	Loại bỏ mặt quay vào trong
K	2
M	0.25

Câu 9

H	Thuật toán tương thích với dòng quét được phát triển bởi
Đ	Cả {T1} và T{2}
T1	Wylie
T2	Evans
T3	Cat mull
K	2
M	0.25

Câu 10

H	“Ray-tracing” (theo dấu tia) là mở rộng của
Đ	Ray casting
T1	Ray calling
T2	Ray sampling
T3	Ray coherence
K	2
M	0.25

Câu 11

H	Z -buffer là thuật toán
Đ	Đơn giản nhất
T1	Phức tạp nhất
T2	Rộng nhất
T3	Không lựa chọn nào
K	2
M	0.25

Câu 12

H	Kiểu cây nào của cấu trúc dữ liệu mà mỗi nút trong có 4 nút con
Đ	Cây tứ phân (Quad tree)
T1	Cây 4 điểm (Point quad tree)
T2	Cây 4 cạnh (Edge quad tree)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 13

H	Thuật toán người thợ sơn dựa trên thuộc tính của
Đ	Bộ đệm khung (Frame buffer)
T1	Đa giác (Polygon)
T2	Bộ đệm độ sâu (Depth buffer)
T3	Không lựa chọn nào
K	2
M	0.25

II.7. Đường cong và bề mặt (14)

Câu 1

H	Tham số chính của thuộc tính đường cong và bề mặt là
Đ	Tất cả các lựa chọn {T1}, {T2} và {T3}
T1	Kiểu (Type)
T2	Độ rộng (Width)
T3	Màu sắc (Color)
K	2
M	0.25

Câu 2

H	Các đường cong mảnh hóa với các độ rộng khác nhau có thể được hiển thị sử dụng
Đ	Các nhịp ngang hoặc dọc (Horizontal and vertical spans)
T1	Các nhịp ngang (Horizontal spans)
T2	Các nhịp dọc (Vertical spans)
T3	Chỉ {T2}
K	2
M	0.25

Câu 3

H	Nếu độ lớn của hệ số góc đoạn cong nhỏ hơn 1 thì
Đ	Chúng ta có thể vẽ các nhịp dọc

T1	Chúng ta có thể vẽ các nhịp ngang
T2	Chỉ {T1}
T3	Tất cả các lựa chọn trên
K	2
M	0.25

Câu 4

H	Nếu hệ số góc là 1 thì đường tròn, elip và các loại đường cong khác sẽ ở dạng
Đ	Nhỏ nhất (Thinnest)
T1	Dày (Thick)
T2	Lớn (Big)
T3	Gồ ghề (Rough)
K	2
M	0.25

Câu 5

H	Một trong các phương pháp hiển thị đường con dày là
Đ	Hệ số góc (Curve slope)
T1	Độ rộng (Curve width)
T2	Đầu mút (Curve cap)
T3	Chỉ {T2}
K	2
M	0.25

Câu 6

H	Mặt nạ điểm ảnh (pixel masks) để thực thi lựa chọn kiểu đường thẳng được sử dụng trong thuật toán sau để tạo ra các mẫu gạch (dashed) hoặc chấm (dotted)
Đ	Thuật toán đường cong màn hình (Raster curve algorithm)
T1	Thuật toán đường thẳng màn hình (Raster line algorithm)
T2	Thuật toán quét màn hình (Raster scan algorithm)
T3	Thuật toán đường cong ngẫu nhiên (Random curve algorithm)
K	2
M	0.25

Câu 7

H	Chúng ta có thể sinh ra các mẫu gạch (dashed) trong các mẫu 1/8 cung tròn (octant), đoạn của hình tròn, đoạn nằm ngang sử dụng
---	--

Đ	Tính đối xứng của đường tròn (Circle symmetry)
T1	Đường tròn (Circles)
T2	Hệ số góc đoạn cong (Curve slope)
T3	Chỉ {T1}
K	2
M	0.25

Câu 8

H	Chức năng của mặt nạ điểm ảnh (pixel mask) là
Đ	Để hiển thị mẫu gạch (dashes) và khoảng trống giữa các mẫu gạch (inter dash spaces) theo hệ số góc
T1	Để hiển thị thuộc tính cong
T2	Để hiển thị các đường cong dày
T3	Không lựa chọn nào
K	2
M	0.25

Câu 9

H	Nếu chúng ta muốn hiển thị các mẫu gạch có độ dài cố định (constant-length dashes), chúng ta cần phải.
Đ	Điều chỉnh số lượng điểm ảnh trong mỗi mẫu gạch
T1	Điều chỉnh số lượng các chấm
T2	Sử dụng các hàm kiểu đường thẳng
T3	Không phải {Đ} cũng không phải {T2}
K	2
M	0.25

Câu 10

H	Các đường cong được hiển thị với các bút hình chữ nhật sẽ
Đ	Mảnh hơn và hệ số góc nhỏ hơn 1 (Thicker and magnitude slope is 1)
T1	Mảnh hơn (Thinner)
T2	Dày hơn và hệ số góc lớn hơn 1 (Thicker and magnitude slope >1)
T3	{Đ} or {T1}
K	2
M	0.25

Câu 11

H	Điền vào phần bỏ trống _____ là một dải linh hoạt (flexible strip) được sử dụng để tạo ra đường cong mượt sử dụng một tập các điểm.
Đ	Đường spline (Spline)

T1	Phương pháp dòng quét (Scan-line method)
T2	Phương pháp sắp xếp độ sâu (Depth-sorting method)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 12

H	Kiểu của đường cong spline là
Đ	Cả {T1} và {T2}
T1	Đường spline mở (Open spline)
T2	Đường spline đóng (Closed spline)
T3	Không lựa chọn nào
K	2
M	0.25

Câu 13

H	Spline bậc 3 là
Đ	Cả {T1} và {T2}
T1	Đơn giản để tính toán
T2	Cung cấp các đường cong liên tục
T3	Không lựa chọn nào
K	2
M	0.25

Câu 14

H	Dạng tham số của đường spline 3D là
Đ	$X=f(t), y=g(t), z=h(t)$
T1	$X=a_0, y=b_0, z=c_0$
T2	$F(t)=0, g(t)=0, h(t)=0$
T3	Không lựa chọn nào
K	2
M	0.25

II.8. Ánh sáng (11)

Câu 1

H	Hai mô hình màu là
Đ	RGB và CMYK
T1	RGB và CMKY
T2	RBG và CYMK
T3	Không lựa chọn nào
K	2

M	0.25
---	------

Câu 2

H	Mô hình RGB được sử dụng cho
Đ	Màn hình máy tính
T1	In ấn
T2	Tô màu
T3	Không lựa chọn nào
K	2
M	0.25

Câu 3

H	Mô hình CMYK được sử dụng cho
Đ	In ấn
T1	Màn hình máy tính
T2	Tô màu
T3	Không lựa chọn nào
K	2
M	0.25

Câu 4

H	Phần giao của 3 màu cơ bản RGB sẽ tạo ra màu
Đ	Trắng
T1	Đen
T2	Tím
T3	Xanh nước biển
K	2
M	0.25

Câu 5

H	Phần giao của 3 màu cơ bản CMYK sẽ tạo ra màu
Đ	Đen
T1	Trắng
T2	Tím
T3	Xanh nước biển
K	2
M	0.25

Câu 6

H	Kỹ thuật tạo bóng nào dưới đây (shading techniques) tạo ra hiệu ứng bề mặt phẳng (faceted appearance)?
Đ	Gouraud
T1	flat

T2	Phong
T3	Không lựa chọn nào
K	2
M	0.25

Câu 7

H	Kỹ thuật tạo bóng nào dưới đây cho phép miêu tả điểm sang (specular highlights)?
Đ	Phong
T1	flat
T2	Gouraud
T3	Không lựa chọn nào
K	2
M	0.25

Câu 8

H	Khẳng định nào dưới đây về các mô hình tạo bóng là đúng
Đ	Khi áp dụng tạo bóng phẳng, mô hình ánh sáng chỉ được áp dụng một lần cho mỗi đa giác, và mọi điểm ảnh trên đa giác đó nhận cùng giá trị cường độ sáng
T1	Tạo bóng Phong giả định cùng một vectơ pháp tuyến được áp dụng cho mọi điểm ảnh trên bề mặt
T2	Tạo bóng Gouraud luôn có điểm sáng
T3	Tạo bóng cố định (Constant) yêu cầu tính toán phức tạp hơn tạo bóng Phong
K	2
M	0.25

Câu 9

H	Trong thuật toán dò tia, nếu ta thêm nhiễu nhỏ vào mỗi tia, điều đó ảnh hưởng thế nào đến hình ảnh kết quả
Đ	Mờ ảnh kết quả
T1	Biến dạng ảnh kết quả
T2	Xuất hiện hạt trong ảnh kết quả
T3	Tăng hiện tượng rang cưa trong ảnh kết quả
K	2
M	0.25

Câu 10

H	Khi sử dụng mô hình phản chiếu Phong (Phong Reflectance Model), việc tính các kênh màu được thực hiện như sau
Đ	các kênh đỏ, xanh lá, xanh dương được tính độc lập.
T1	các kênh đỏ, xanh lá, xanh dương được tính cùng nhau.
T2	các kênh đỏ, xanh lá được tính độc lập.

T3	các xanh lá, xanh dương được tính độc lập.
K	2
M	0.25

Câu 11

H	Mô hình tạo bóng nào yêu cầu tính toán nhiều hơn: mô hình tạo bóng Phong, mô hình tạo bóng Gouraud, mô hình tạo bóng phẳng
Đ	mô hình tạo bóng Phong
T1	mô hình tạo bóng Gouraud
T2	mô hình tạo bóng phẳng
T3	Các mô hình có lượng tính toán như nhau
K	2
M	0.25

II. 9 OpenGL

Câu 1

H	Ý nghĩa của i,f,v,d... theo qui ước trong các hàm của OpenGL?
Đ	Chỉ rõ kiểu dữ liệu (int, float, pointer, double, ...)
T1	Không có ý nghĩa gì đặc biệt
T2	Chỉ rõ kiểu dữ liệu trong OpenGL được viết cho CPU
T3	Đặc tả đầu ra (Ví dụ: immediate mode, frame buffer, virtual mode, double-buffer, ...)
K	2
M	0.25

Câu 2

H	Đâu là 2 lựa chọn của glShadeModel()?
Đ	SMOOTH và FLAT
T1	DEPTH và STENCIL
T2	WIDE và NARROW
T3	FOREGROUND và BACKGROUND
K	2
M	0.25

Câu 3

H	Cỡ của ma trận biến đổi 3 chiều
Đ	4×4
T1	2×2
T2	3×3
T3	3×4

K	2
M	0.25

Câu 4

H	Bạn có thể chỉ rõ vị trí của một nguồn sáng trong OpenGL?
Đ	Có
T1	Không
T2	Thỉnh thoảng
T3	Có bởi người sử dụng
K	2
M	0.25

Câu 5

H	VA hoặc VAO viết tắt cho?
Đ	Vertex Array Object
T1	Vector Array Object
T2	Vertex Automation Output
T3	Vector Array Output
K	2
M	0.25

Câu 6

H	Chúng ta phải làm gì để quay đối tượng quanh một điểm không phải gốc tọa độ?
Đ	Dịch đối tượng tới gốc tọa độ, xoay quanh gốc tọa độ sau đó dịch tới vị trí ban đầu
T1	Phép quay chỉ thực hiện được quanh gốc tọa độ
T2	Thực hiện câu lệnh glRotate và chỉ rõ điểm mà đối tượng quay quanh
T3	Thực hiện câu lệnh glTranslate
K	2
M	0.25

Câu 7

H	Trong OpenGL, GLSL là viết tắt của?
Đ	Graphics Library Shader Language
T1	Graphical Library of Shader Languages
T2	Geographic Land and Survey Library
T3	Graphical Language and Shading Library
K	2
M	0.25

Câu 8

H	glMaterialf(...) thực hiện chức năng?
Đ	Định nghĩa thuộc tính chất liệu của đối tượng được vẽ cho hiệu ứng hiển thị
T1	Bật thuộc tính chất liệu

T2	Chuyển sang chế độ chất liệu để thêm hiệu ứng hiển thị
T3	Lấy thuộc tính chất liệu của đối tượng được vẽ
K	2
M	0.25

Câu 9

H	Hai tham số cho hàm glShadeModel()?
Đ	SMOOTH và FLAT
T1	DEPTH và STENCIL
T2	WIDE và NARROW
T3	FOREGROUND và BACKGROUND
K	2
M	0.25

Câu 10

H	Dạng đa giác nào hoạt động tốt nhất trong VBO để mô hình hóa đối tượng đặc (solid)?
Đ	Triangles, Triangle Strips và Triangle Fans
T1	Triangles, quads và N-gons
T2	Quads, N-gons và unilateral N-gons
T3	N-gonal coplanar line strips
K	2
M	0.25

Câu 11

H	Các thành phần tạo nên nguồn sáng trong OpenGL?
Đ	Specular và Ambient.
T1	Diffuse, Specular, và Ambient.
T2	Diffuse và Ambient.
T3	Diffuse, Opaque, Ambient.
K	2
M	0.25

Câu 12

H	Loại dữ liệu nào được lưu trữ trong VBO?
Đ	Đỉnh đa giác (Vertices)
T1	Thông tin chia sẻ ngữ cảnh (Context-sharing Information)
T2	Điểm ảnh Pixels
T3	Phân mảnh (Fragments)
K	2
M	0.25

Câu 13

H	Hằng số hợp lệ cho glStencilOp?
Đ	GL_KEEP, GL_ZERO, GL_REPLACE, GL_INCR, GL_INCR_WRAP, GL_DECR, GL_DECR_WRAP, GL_INVERT
T1	Một số nằm trong khoảng 0 và 255
T2	TRUE hoặc FALSE
T3	GL_R, GL_G, GL_B, GL_RGB, GL_RGBA, GL_ARGB, GL_BGRA
K	2
M	0.25

Câu 14

H	Biến đổi (transformation) là gì ?
Đ	Phép toán được sử dụng để tạo các di chuyển của các điểm và đối tượng theo mong muốn
T1	Chuyển qua lại giữa các bộ đệm
T2	Hàm của OpenGL dùng để biến dạng (morph) đa giác thành đa giác khác.
T3	Phép biến hình
K	2
M	0.25

Câu 15

H	Nói chung, các đa giác nguyên thủy (primitive polygon) nào được dùng để tạo lưới để biểu diễn các đối tượng phức tạp
Đ	Tam giác (Triangle)
T1	Hình vuông (Square)
T2	Hình tròn (Circle)
T3	Hình chữ nhật (Rectangle)
K	2
M	0.25

Câu 16

H	Nếu không bật nguồn sáng, hàm nào dưới đây chỉ rõ màu của các đỉnh
Đ	glColor()
T1	glClearColor()
T2	glDisplayfunc()
T3	Không hàm nào
K	2
M	0.25

Câu 17

H	Sự khác nhau giữa glColor3d và glColor3f?
Đ	glColor3d có tham số kiểu double trong khi glColor3f có tham số kiểu float
T1	glColor3d chỉ thiết lập RGB, trong khi glColor3f thiết lập R,G,B và A
T2	glColor3d cho phép thao tác với màu 3d trong khi glColor3f chỉ cho phép 8-bit màu
T3	glColor3d trong không gian thực trong khi glColor3f trong không gian số nguyên
K	2
M	0.25

Câu 18

H	FBO là gì?
Đ	Framebuffer Object
T1	File Buffered Output
T2	Frictionless Baryonic Oscillation
T3	Friday Buy Out
K	2
M	0.25

Câu 19

H	Hàm nào có thể thiết lập kích cỡ của vùng hiển thị
Đ	glViewport()
T1	gluPerspective()
T2	Không lựa chọn nào
T3	glDisplayfunc()
K	2
M	0.25

Câu 20

H	Ánh sáng môi trường (ambient light) có trong OpenGL?
Đ	Có
T1	Thỉnh thoảng
T2	Không
T3	Tùy từng phiên bản
K	2
M	0.25

Câu 21

H	OpenGL có hỗ trợ màn hình hóa (Rasterization)?
Đ	Có

T1	Không
T2	Thỉnh thoảng
T3	Tùy từng phiên bản
K	2
M	0.25

Câu 22

H	OpenGL là viết tắt của
Đ	Open Graphics Library
T1	Open General Liability
T2	Open Guide Line
T3	Open Graphics Layer
K	2
M	0.25

Câu 23

H	Opengl dùng hệ trục tọa độ nào?
Đ	Hệ tọa độ Decac ba chiều vuông góc
T1	Hệ tọa độ Decac hai chiều vuông góc
T2	Hệ tọa độ cực ba chiều vuông góc
T3	Hệ tọa độ cực hai chiều vuông góc
K	2
M	0.25

Câu 24

H	Nếu có phép quay góc α . Phép quay ngược là
Đ	Phép quay góc $-\alpha$
T1	Phép quay góc 2α
T2	Phép quay 0^0
T3	
K	2
M	0.25

Câu 25

H	Câu lệnh dùng để vẽ các hình nguyên thủy trong OpenGL (OpenGL primitives).
Đ	glVertex
T1	glColor
T2	glEnable
T3	glShadeModel
K	2
M	0.25

Câu 26

H	Loại dữ liệu nào được lưu trữ trong VAO?
Đ	Trạng thái của đỉnh (Vertex state) và các thông tin dữ liệu khác dùng cho rendering trong luồng xử lý đồ họa
T1	Các thông tin lũy thừa để phân tích dữ liệu dạng sóng
T2	Thông tin tài nguyên của chương trình như trees, lists và data types
T3	Các chuỗi nhị phân
K	2
M	0.25

Câu 27

H	VBO là gì?
Đ	Vertex Buffer Object
T1	Vertex Binding Object
T2	Variable Buffer Output
T3	Vertex Buffer Output
K	2
M	0.25

Câu 28

H	“depth buffer” là gì và chức năng của nó?
Đ	Một kiểu bộ đệm tích lũy thông tin độ sâu trong khung cảnh 3 chiều
T1	Một lược đồ tổ chức cảnh 3 chiều
T2	Một bộ đệm được đặc tả để thực hiện cảnh dưới nước
T3	Một bộ đệm chứa thông tin màu sắc
K	2
M	0.25

Câu 29

H	OpenGL có cung cấp các đối tượng nguyên thủy như cube, cone, pyramid?
Đ	Không chính thức trong đặc tả của OpenGL nhưng các đối tượng này được cung cấp bởi thư viện GLUT
T1	Có nhưng phải dùng glEnable(GL_PRIMITIVES) trước câu lệnh glBegin
T2	Có.
T3	Không bao giờ
K	2
M	0.25

Câu 30

H	OpenGL sử dụng mô hình client-server và một máy trạng thái (state machine)?
Đ	Có
T1	Không liên quan
T2	Không
T3	Thỉnh thoảng
K	2
M	0.25

Câu 31

H	OpenGL có một máy quay (camera) chính thức không?
Đ	Không
T1	Có
T2	Phải chỉ rõ khi thiết lập môi trường lập trình
T3	Có nếu người sử dụng bật lên
K	2
M	0.25

Câu 32

H	Tổ chức chịu trách nhiệm cho các chuẩn và phát triển của OpenGL?
Đ	Khronos
T1	Crysis
T2	Cryptic
T3	Chronos
K	2
M	0.25

Câu 33

H	Mục đích của GL_REPEAT trong OpenGL?
Đ	Lặp lại ảnh chất liệu theo một hướng (ngang hoặc dọc)
T1	Lặp lại câu lệnh cuối cùng
T2	Kích hoạt chế độ 2 bộ đệm
T3	Khởi động lại ứng dụng
K	2
M	0.25

Câu 34

H	Để có được tính toán chính xác cho việc tạo bóng. Độ dài cho các vecto pháp tuyến của đối tượng là?
Đ	1 (đơn vị độ dài)
T1	2

T2	Bất cứ giá trị nào.
T3	Phụ thuộc vào kích cỡ của đối tượng được vẽ
K	2
M	0.25

Câu 35

H	Tại sao phải xóa bộ đệm?
Đ	Khởi tạo lại một trạng thái
T1	Yêu cầu bộ đệm xuất dữ liệu ra màn hình
T2	Để hoán đổi ộ đệm
T3	Để hiển thị được rõ ràng hơn
K	2
M	0.25

Câu 36

H	glViewport dùng để làm gì?
Đ	Thiết lập sự mở rộng cho khung nhìn hiện tại
T1	Thiết lập một viewport hoạt động
T2	Yêu cầu dùng cho hàm glOrtho
T3	Tắt chế độ tích lũy và kích hoạt viewport chuẩn.
K	2
M	0.25

Câu 37

H	Hai kiểu của shader?
Đ	Vertex và fragment.
T1	Fragment và polygon.
T2	Vertex và polygon
T3	Polygon và particle
K	2
M	0.25

Câu 38

H	Các hàm trộn (blending function) không phụ thuộc thứ tự
Đ	Sai
T1	Đúng
T2	Thỉnh thoảng
T3	Do người lập trình lựa chọn
K	2
M	0.25

Câu 39

H	X,Y,Z,S,T,U và V là gì?
Đ	Thành phần tọa độ (Coordinate components)
T1	Các ký tự trong bảng chữ cái và không có ý nghĩa
T2	Tham số lựa chọn trong hàm glVertex*
T3	Biến của OpenGL
K	2
M	0.25

Câu 40

H	OpenGL có cung cấp hiệu ứng vật lý (physics), hệ thống hạt (particle systems)?
Đ	Không.
T1	Có nhưng trong thư viện mở rộng Box2d
T2	Có nhưng tùy phiên bản
T3	Chỉ cung cấp hiệu ứng vật lý
K	2
M	0.25

Câu 41

H	Các giá trị “R”, “G”, “B” và “A” trong ngữ cảnh màu sắc có ý nghĩa gì?
Đ	Các kênh Red, Green, Blue và Alpha tương ứng
T1	Rotate, Gyrate, Blend và Amorphize
T2	Red, Green, Black, Auburn
T3	Các giá trị “R”, “G”, “B” và “A” trong ngữ cảnh màu sắc có ý nghĩa gì?
K	2
M	0.25

Câu 42

H	OpenGL có thực hiện các thủ tục cắt xén không
Đ	Có
T1	Không
T2	Có nhưng tùy phiên bản
T3	Chỉ cắt xen đoạn thẳng
K	2
M	0.25

Câu 43

H	Loại mảng đỉnh (vertex array) nào không được cung cấp bởi OpenGL?
Đ	Đa giác (Polygons)
T1	Đỉnh (Vertices)

T2	Màu sắc (Colors)
T3	Vecto pháp tuyến (Normals)
K	2
M	0.25

III. Phần tự luận

III.1. Các thuật toán tô màu

Câu 1 (1 điểm)

H	<p>(1 điểm) Trình bày thuật toán tô màu loang 4 láng giềng. Áp dụng liệt kê thứ tự tô màu của các ô trong hình vẽ dưới đây với ô được đánh số 1 là seed.</p>
Đ	<pre>void boundaryFill4 (int x, int y, int fill, int boundary) { int current; current = getpixel (x, y); if ((current != boundary) && (current != fill)) { setcolor (fill); setpixel (x, y); boundaryFill4 (x+1, y, fill, boundary); boundaryFill4 (x-1, y, fill, boundary); boundaryFill4 (x, y+1, fill, boundary); boundaryFill4 (x, y-1, fill, boundary); } }</pre>

K	1
M	1

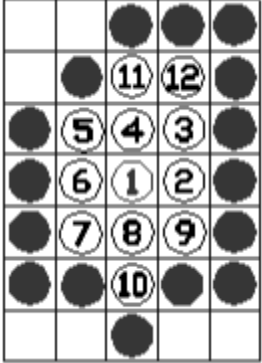
Câu 2 (1 điểm)

H	<p>(1 điểm) Trình bày thuật toán tô màu loang 4 láng giềng. Áp dụng liệt kê thứ tự tô màu của các ô trong hình vẽ dưới đây với ô được đánh số 1 là seed.</p>
Đ	<pre>void boundaryFill4 (int x, int y, int fill, int boundary) { int current; current = getpixel (x, y); if ((current != boundary) && (current != fill)) { setcolor (fill); setpixel (x, y); boundaryFill4 (x+1, y, fill, boundary); } }</pre>

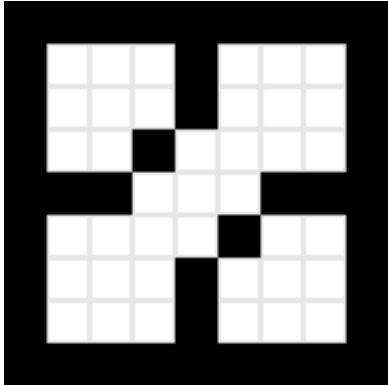
	<pre> boundaryFill4 (x-1, y, fill, boundary); boundaryFill4 (x, y+1, fill, boundary); boundaryFill4 (x, y-1, fill, boundary); } } </pre>
K	1
M	1

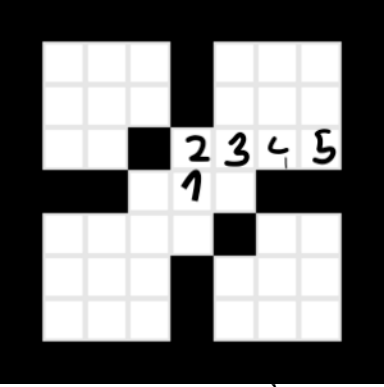
Câu 3 (1 điểm)

H	<p>(1 điểm) Trình bày thuật toán tô màu loang 4 láng giềng. Áp dụng liệt kê thứ tự tô màu của các ô trong hình vẽ dưới đây với ô được đánh số 1 là seed.</p>
Đ	<pre> void boundaryFill4 (int x, int y, int fill, int boundary) { int current; current = getpixel (x, y); </pre>

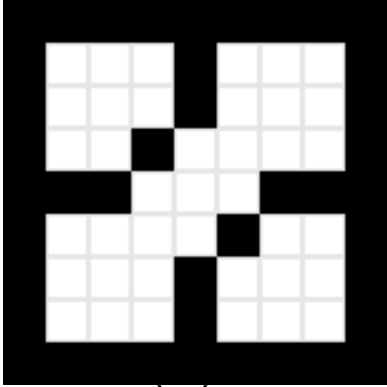
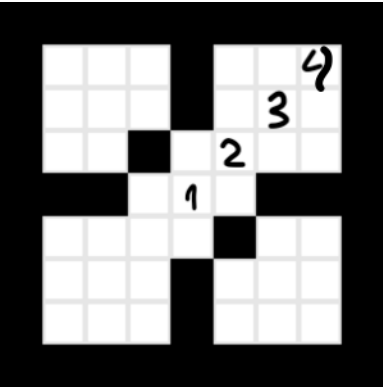
	<pre> if ((current != boundary) && (current != fill)) { setcolor (fill); setpixel (x, y); boundaryFill4 (x+1, y, fill, boundary); boundaryFill4 (x-1, y, fill, boundary); boundaryFill4 (x, y+1, fill, boundary); boundaryFill4 (x, y-1, fill, boundary); } </pre> 
K	1
M	1

Câu 4 (1 điểm)

H	<p>a. (0.5 điểm) Trình bày thuật toán tô màu loang 4 láng giềng. Áp dụng liệt kê thứ tự tô màu của các ô trong hình vẽ dưới đây (SV tự lựa chọn ô xuất phát ban đầu)</p>  <p>b. (0.5 điểm) Nêu sự khác biệt về kết quả khi áp dụng thuật toán tô phủ loang 4 láng giềng và 8 láng giềng.</p>
Đ	a.

	 <p>b. Tô phủ 4 láng giềng sẽ không hoàn thiện hết phần góc trên bên trái và góc dưới bên phải</p>
K	1
M	1

Câu 5 (1 điểm)

H	<p>a. (0.5 điểm) Trình bày thuật toán tô màu loang 8 láng giềng. Áp dụng liệt kê thứ tự tô màu của các ô trong hình vẽ dưới đây (SV tự lựa chọn ô xuất phát ban đầu)</p>  <p>b. (0.5 điểm) Nêu sự khác biệt về kết quả khi áp dụng thuật toán tô phủ loang 4 láng giềng và 8 láng giềng.</p>
Đ	<p>a.</p> 

	b. Tô phủ 4 lát giềng sẽ không hoàn thiện hết phần góc trên bên trái và góc dưới bên phải
K	1
M	1

Câu 6 (1 điểm)

H	(1 điểm) Xét đường thẳng đi qua 2 điểm (1, 2) và (2, 4). (a) (0.5 điểm) Viết phương trình đường thẳng dùng hệ số góc $y=ax+b$ (explicit equation). (b) (0.25 điểm) Viết phương trình dạng tổng quát của đường thẳng. Xác định vecto pháp tuyến của đường thẳng (c) (0.25 điểm) Viết phương trình đường thẳng dưới dạng tham số. Xác định vecto chỉ phương của đường thẳng
Đ	a. $y=2x$ b. $2x-y=0$, $n=(2,-1)$ c. $x=1+t$ $y=2+2t$ $v=(1,2)$
K	2
M	1

III.2 Các thuật toán vẽ đường thẳng/ vẽ đường tròn




Câu 1 (1 điểm)

H	(1 điểm) Trình bày thuật toán Bresenham. Áp dụng vẽ một đường thẳng từ điểm P1(0, 0) tới điểm P2(10, 6) dùng thuật toán Bresenham.
Đ	<pre> Void Bresenham (x1,y1, x2,y2) { int x = x1, y = y1; int dx = x2 - x1, dy = y2 - y1, dT = 2(dy - dx), dS = 2dy; int d = 2dy - dx; setPixel(x, y); while (x < x2) { x++; if (d < 0) d = d + dS; else { y++; d = d + dT; } setPixel(x, y); } } </pre>

	<pre> d = d + dT; } setPixel(x, y); }</pre>																																																																		
	<table><tr><td colspan="11">x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)</td></tr><tr><td colspan="10">while (x < x')</td><td>while (x ≥ x')</td></tr><tr><td>X</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr><tr><td>Y</td><td>1</td><td>x</td><td>2</td><td>x</td><td>3</td><td>4</td><td>x</td><td>5</td><td>x</td><td>6</td></tr><tr><td>D</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td></tr><tr><td>(x,y)</td><td>(1,2)</td><td>(2,1)</td><td>(3,2)</td><td>(4,2)</td><td>(5,3)</td><td>(6,4)</td><td>(7,4)</td><td>(8,5)</td><td>(9,5)</td><td>(10,6)</td></tr></table>	x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)											while (x < x')										while (x ≥ x')	X	1	2	3	4	5	6	7	8	9	10	Y	1	x	2	x	3	4	x	5	x	6	D	-6	6	-2	10	2	-6	6	-2	10	2	(x,y)	(1,2)	(2,1)	(3,2)	(4,2)	(5,3)	(6,4)	(7,4)	(8,5)	(9,5)	(10,6)
x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)																																																																			
while (x < x')										while (x ≥ x')																																																									
X	1	2	3	4	5	6	7	8	9	10																																																									
Y	1	x	2	x	3	4	x	5	x	6																																																									
D	-6	6	-2	10	2	-6	6	-2	10	2																																																									
(x,y)	(1,2)	(2,1)	(3,2)	(4,2)	(5,3)	(6,4)	(7,4)	(8,5)	(9,5)	(10,6)																																																									
K	1																																																																		
M	1																																																																		

Câu 2 (1 điểm)

H	(1 điểm) Trình bày thuật toán điểm giữa. Áp dụng vẽ một đường thẳng từ điểm P1(0, 0) tới điểm P2(10, 6) dùng thuật toán điểm giữa.
Đ	<pre> MiddlePoint(x0,y0,x1,y1) M >1 { dx=x1-x0; dy=y1-y0; x=x0; y=y0 plot(x,y); while (y<y1) { y=y+1; if (d<0) d=d+dx; else { d=d+dx-dy x=x+1 } plot(x,y) } } MiddlePoint(x0,y0,x1,y1) M <1 { </pre>

	<pre>dx=x1-x0; dy=y1-y0; x=x0; y=y0 plot(x,y); while (x<x1) { x=x+1; if (d<0) d=d+dx; else { d=d+dx-dy y=y+1 } plot(x,y) }</pre>																																																																				
	<table><tr><td colspan="11">$x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, \text{setPixel}(0,0)$</td></tr><tr><td colspan="11">while ($x < x_1$)</td><td>while ($x \geq x_1$)</td></tr><tr><td>X</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td rowspan="4"></td></tr><tr><td>Y</td><td>1</td><td>x</td><td>2</td><td>x</td><td>3</td><td>4</td><td>x</td><td>5</td><td>x</td><td>6</td></tr><tr><td>D</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td></tr><tr><td>(x,y)</td><td>$(1,2)$</td><td>$(2,1)$</td><td>$(3,2)$</td><td>$(4,2)$</td><td>$(5,3)$</td><td>$(6,4)$</td><td>$(7,4)$</td><td>$(8,5)$</td><td>$(9,5)$</td><td>$(10,6)$</td></tr></table>	$x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, \text{setPixel}(0,0)$											while ($x < x_1$)											while ($x \geq x_1$)	X	1	2	3	4	5	6	7	8	9	10		Y	1	x	2	x	3	4	x	5	x	6	D	-6	6	-2	10	2	-6	6	-2	10	2	(x,y)	$(1,2)$	$(2,1)$	$(3,2)$	$(4,2)$	$(5,3)$	$(6,4)$	$(7,4)$	$(8,5)$	$(9,5)$	$(10,6)$
$x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, \text{setPixel}(0,0)$																																																																					
while ($x < x_1$)											while ($x \geq x_1$)																																																										
X	1	2	3	4	5	6	7	8	9	10																																																											
Y	1	x	2	x	3	4	x	5	x	6																																																											
D	-6	6	-2	10	2	-6	6	-2	10	2																																																											
(x,y)	$(1,2)$	$(2,1)$	$(3,2)$	$(4,2)$	$(5,3)$	$(6,4)$	$(7,4)$	$(8,5)$	$(9,5)$	$(10,6)$																																																											
K	1																																																																				
M	1																																																																				

Câu 3 (1 điểm)

	(1 điểm) Trình bày thuật toán DDA. Áp dụng vẽ một đường thẳng từ điểm P1(0, 0) tới điểm P2(10, 6) dùng thuật toán DDA.
Đ	<pre> Void DDA(x0,y0,x1,y1) { Tính m; If m<1 { Float y= y0; </pre>

	<pre>For (x=x0;x<=x1;x++, y+=m) setPixel(x,round(y)); } Else { Float x= x0; For (y=y0;y<=y1;y++, x+=m) setPixel(round(x), y); }</pre>																																																																				
	<table><tr><td colspan="11">x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)</td></tr><tr><td colspan="11">while (x< x'2)</td><td>while (x≥x'2)</td></tr><tr><td>X</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td rowspan="4"></td></tr><tr><td>Y</td><td>1</td><td>x</td><td>2</td><td>x</td><td>3</td><td>4</td><td>x</td><td>5</td><td>x</td><td>6</td></tr><tr><td>D</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td><td>-6</td><td>6</td><td>-2</td><td>10</td><td>2</td></tr><tr><td>(x,y)</td><td>(1,2)</td><td>(2,1)</td><td>(3,2)</td><td>(4,2)</td><td>(5,3)</td><td>(6,4)</td><td>(7,4)</td><td>(8,5)</td><td>(9,5)</td><td>(10,6)</td></tr></table>	x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)											while (x< x'2)											while (x≥x'2)	X	1	2	3	4	5	6	7	8	9	10		Y	1	x	2	x	3	4	x	5	x	6	D	-6	6	-2	10	2	-6	6	-2	10	2	(x,y)	(1,2)	(2,1)	(3,2)	(4,2)	(5,3)	(6,4)	(7,4)	(8,5)	(9,5)	(10,6)
x=0, y=0, dx=10-0=10, dy=6-0=6, dT=2*(6-10)=-8, dS=2*6=12, d=2*6-10=2, setPixel(0,0)																																																																					
while (x< x'2)											while (x≥x'2)																																																										
X	1	2	3	4	5	6	7	8	9	10																																																											
Y	1	x	2	x	3	4	x	5	x	6																																																											
D	-6	6	-2	10	2	-6	6	-2	10	2																																																											
(x,y)	(1,2)	(2,1)	(3,2)	(4,2)	(5,3)	(6,4)	(7,4)	(8,5)	(9,5)	(10,6)																																																											
K	1																																																																				
M	1																																																																				

Câu 4 (1 điểm)

H	(1 điểm) Trình bày thuật toán Bresenham. Áp dụng vẽ một đường thẳng từ điểm P1(0, 0) tới điểm P2(5, 3) dùng thuật toán Bresenham.
Đ	<pre> Void Bresendham (x1,y1, x2,y2) { int x = x1, y =y1; int dx = x2 - x1, dy = y2 - y1, dT = 2(dy - dx), dS = 2dy; int d = 2dy - dx; setPixel(x, y); while (x < x2) { x++; if (d < 0) d = d + dS; else { y++; d = d + dT; } setPixel(x, y); } } </pre>

		<table><tr><th>(x, y)</th><th>$\bar{\epsilon}$</th><th>description</th></tr><tr><td>(0, 0)</td><td>-2</td><td>illuminate pixel (0, 0)</td></tr><tr><td></td><td>1</td><td>increment ϵ by Δy</td></tr><tr><td>(1, 0)</td><td></td><td>increment x by 1</td></tr><tr><td>(1, 0)</td><td>1</td><td>illuminate pixel (1, 0)</td></tr><tr><td></td><td></td><td>since $\bar{\epsilon} > 0$</td></tr><tr><td>(1, 1)</td><td></td><td>increment y by 1</td></tr><tr><td></td><td>-4</td><td>decrement $\bar{\epsilon}$ by 5</td></tr><tr><td></td><td>-1</td><td>increment ϵ by Δy</td></tr><tr><td>(2, 1)</td><td></td><td>increment x by 1</td></tr><tr><td>(2, 1)</td><td>-1</td><td>illuminate pixel (2, 1)</td></tr><tr><td></td><td>2</td><td>increment ϵ by Δy</td></tr><tr><td>(3, 1)</td><td></td><td>increment x by 1</td></tr><tr><td>(3, 1)</td><td>2</td><td>illuminate pixel (3, 1)</td></tr><tr><td></td><td></td><td>since $\bar{\epsilon} > 0$</td></tr><tr><td>(3, 2)</td><td></td><td>increment y by 1</td></tr><tr><td></td><td>-3</td><td>decrement $\bar{\epsilon}$ by 5</td></tr><tr><td></td><td>0</td><td>increment ϵ by Δy</td></tr><tr><td>(4, 2)</td><td></td><td>increment x by 1</td></tr><tr><td>(4, 2)</td><td>0</td><td>illuminate pixel (4, 2)</td></tr></table>	(x, y)	$\bar{\epsilon}$	description	(0, 0)	-2	illuminate pixel (0, 0)		1	increment ϵ by Δy	(1, 0)		increment x by 1	(1, 0)	1	illuminate pixel (1, 0)			since $\bar{\epsilon} > 0$	(1, 1)		increment y by 1		-4	decrement $\bar{\epsilon}$ by 5		-1	increment ϵ by Δy	(2, 1)		increment x by 1	(2, 1)	-1	illuminate pixel (2, 1)		2	increment ϵ by Δy	(3, 1)		increment x by 1	(3, 1)	2	illuminate pixel (3, 1)			since $\bar{\epsilon} > 0$	(3, 2)		increment y by 1		-3	decrement $\bar{\epsilon}$ by 5		0	increment ϵ by Δy	(4, 2)		increment x by 1	(4, 2)	0	illuminate pixel (4, 2)
(x, y)	$\bar{\epsilon}$	description																																																												
(0, 0)	-2	illuminate pixel (0, 0)																																																												
	1	increment ϵ by Δy																																																												
(1, 0)		increment x by 1																																																												
(1, 0)	1	illuminate pixel (1, 0)																																																												
		since $\bar{\epsilon} > 0$																																																												
(1, 1)		increment y by 1																																																												
	-4	decrement $\bar{\epsilon}$ by 5																																																												
	-1	increment ϵ by Δy																																																												
(2, 1)		increment x by 1																																																												
(2, 1)	-1	illuminate pixel (2, 1)																																																												
	2	increment ϵ by Δy																																																												
(3, 1)		increment x by 1																																																												
(3, 1)	2	illuminate pixel (3, 1)																																																												
		since $\bar{\epsilon} > 0$																																																												
(3, 2)		increment y by 1																																																												
	-3	decrement $\bar{\epsilon}$ by 5																																																												
	0	increment ϵ by Δy																																																												
(4, 2)		increment x by 1																																																												
(4, 2)	0	illuminate pixel (4, 2)																																																												
K	1																																																													
M	1																																																													

Câu 5 (1 điểm)

H	(1 điểm) Trình bày thuật toán DDA. Áp dụng vẽ một đường thẳng từ điểm P1(0,0) tới điểm P2(4, 2) dùng thuật toán DDA.
Đ	<pre> Void DDA(x0,y0,x1,y1) { Tính m; If m<1 { Float y= y0; For (x=x0;x<=x1;x++, y+=m) setPixel(x,round(y)); } Else { Float x= x0; } } </pre>

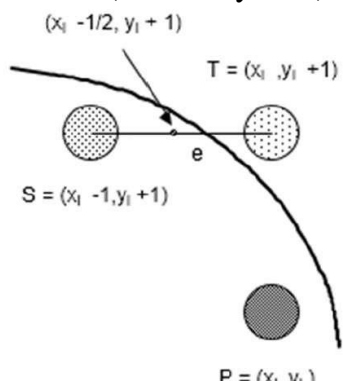
	<pre> For (y=y0;y<=y1;y++, x+=m) setPixel(round(x), y); } dx= x2 - x1 = 4-3 = 1 dy= y2 - y1 = 7-2 = 5 dx < dy thì length = y2-y1 = 5 dx = (x2-x1)/ length = 1/5 =0.2 dy = (y2-y1)/ length = 5/5 = 1 } (0,0), (1,0), (1,1), (2,1), (3,1), (3,2), (4,2) </pre>
K	1
M	1

Câu 6 (1 điểm)

H	(1 điểm) Trình bày thuật toán DDA. Áp dụng vẽ một đường thẳng từ điểm P1(3, 2) tới điểm P2(4, 7) dùng thuật toán DDA.
Đ	<pre> Void DDA(x0,y0,x1,y1) { Tính m; If m<1 { Float y= y0; For (x=x0;x<=x1;x++, y+=m) setPixel(x,round(y)); } Else { Float x= x0; For (y=y0;y<=y1;y++, x+=m) setPixel(round(x), y); } dx= x2 - x1 = 4-3 = 1 dy= y2 - y1 = 7-2 = 5 dx < dy thì length = y2-y1 = 5 dx = (x2-x1)/ length = 1/5 =0.2 dy = (y2-y1)/ length = 5/5 = 1 } } </pre>

	(3,2), (3,3), (3,4), (4,5), (4,6), (4,6)
K	1
M	1

Câu 7 (1 điểm)

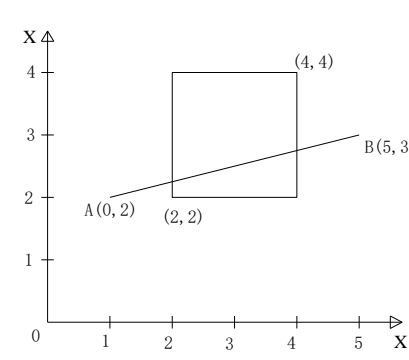
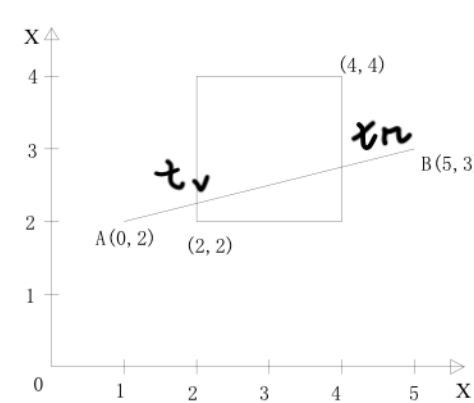
H	(1 điểm) Trình bày thuật toán điểm giữa để vẽ đường tròn
Đ	<p>Đầu vào bán kính r và tâm (x_c, y_c)</p> <p>Bước 1 – điểm đầu tiên</p> <p>$(x_0, y_0) = (0, r)$</p> <p>Bước 2 – Tính giá trị khởi tạo cho tham số quyết định</p> <p>$P_0 P_0 = 5/4 - r$</p> <p>$f(x, y) = x^2 + y^2 - r^2 = 0$</p> <p>$f(x_i - 1/2 + e, y_i + 1)$</p> $= (x_i - 1/2 + e)^2 + (y_i + 1)^2 - r^2$ $= (x_i - 1/2)^2 + (y_i + 1)^2 - r^2 + 2(x_i - 1/2)e + e^2$ $= f(x_i - 1/2, y_i + 1) + 2(x_i - 1/2)e + e^2 = 0$  <p>Gọi $d_i = f(x_i - 1/2, y_i + 1) = -2(x_i - 1/2)e - e^2$</p> <p>Do đó,</p> <p>Nếu $e < 0$ thì $d_i > 0$ vậy ta lựa chọn $S = (x_i - 1, y_i + 1)$.</p> $d_{i+1} = f(x_i - 1 - 1/2, y_i + 1 + 1) = ((x_i - 1/2) - 1)^2 + ((y_i + 1) + 1)^2 - r^2$ $= d_i - 2(x_i - 1) + 2(y_i + 1) + 1$ $= d_i + 2(y_i + 1 - x_i + 1) + 1$ <p>Nếu $e \geq 0$ thì $d_i \leq 0$ vậy ta lựa chọn $T = (x_i, y_i + 1)$</p> $d_{i+1} = f(x_i - 1/2, y_i + 1 + 1)$ $= d_i + 2y_i + 1 + 1$

	<p>Giá trị khởi tạo của di là</p> $d_0 = f(r - 1/2, 0 + 1) = (r - 1/2)^2 + 1^2 - r^2$ $= 5/4 - r \quad \{ 1-r \text{ có thể dùng nếu } r \text{ là nguyên} \}$ <p>Khi S = (xi - 1, yi + 1) được chọn thì</p> $d_{i+1} = d_i + -2x_{i+1} + 2y_{i+1} + 1$ <p>Khi T = (xi, yi + 1) được chọn thì</p> $d_{i+1} = d_i + 2y_{i+1} + 1$ <p>Bước 3 – Tại mỗi bước theo k với giá trị ban đầu K=0, thực hiện kiểm tra sau</p> <p>Nếu PK < 0 thì điểm tiếp theo là (XK+1,YK) của đường tròn tâm (0,0) và</p> $PK+1 = PK + 2XK+1 + 1$ <p>Ngược lại</p> $PK+1 = PK + 2XK+1 + 1 - 2YK+1$ <p>Trong đó, $2XK+1 = 2XK+2$ và $2YK+1 = 2YK-2$.</p> <p>Bước 4 – Xác định các điểm đối xứng còn lại của đường tròn</p> <p>Bước 5 – Dịch chuyển mỗi điểm (X, Y) tính được tới đường tròn tâm (XC,YC) và vẽ điểm đó.</p> $X = X + XC, \quad Y = Y + YC$ <p>Bước 6 – Lặp lại bước 3 đến bước 5 cho đến khi X >= Y.</p>
K	1
M	1

III.3. Các thuật toán cắt xén

Câu 1 (1 điểm)

H	(1 điểm) Sử dụng thuật toán Liang-barsky để cắt xén đường AB trong hình sau
---	---

	
Đ	<p>Tìm tv và tr</p>  <p>Sắp xếp được $0 < tv < tr < 1$ Kết quả $[tv, tr]$</p>
K	2
M	1

Câu 2 (1 điểm)

H	(1 điểm) Trình bày thuật toán Cohend – Sutherland
Đ	
K	2
M	1

Câu 3 (1 điểm)

H	(1 điểm) Trình bày ý tưởng thuật toán Cyrus&Beck
Đ	<p>Tìm tv và tr Sắp xếp các tv và tr theo chiều tăng Kết quả tìm các đoạn $[tv, tr]$</p>

	<p>For (với mỗi cạnh của cửa sổ)</p> <p>+ Lưu đồ:</p>
K	2
M	1

Câu 4 (1 điểm)

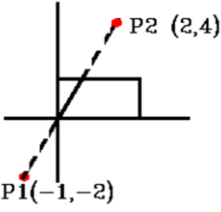
H	(1 điểm) Trình bày một toán toán cắt xén đa giác mà các anh chị biết.
Đ	<p>For (Với mỗi cạnh của đa giác)</p> <p>- Xử lý từng cạnh theo 1 trong 4 trường hợp sau</p>
K	2
M	1

Câu 5 (1 điểm)

H	<p>Tìm đoạn thẳng kết quả (cụ thể) khi áp dụng thuật toán cắt xén Cohend-Sutherland) với các đường thẳng và cửa sổ được cho dưới đây</p> <p>Input : Cửa sổ cắt xén sau</p> $x_{\min} = 4, y_{\min} = 4, x_{\max} = 10, y_{\max} = 8$ <p>Tập các đường thẳng cần cắt xén</p> <p>line 1 : $x_1 = 5, y_1 = 5, x_2 = 7, y_2 = 7$</p> <p>Line 2 : $x_1 = 7, y_1 = 9, x_2 = 11, y_2 = 4$</p> <p>Line 2 : $x_1 = 1, y_1 = 5, x_2 = 4, y_2 = 1$</p>
Đ	<p>Line 1 : Các điểm được giữ lại nằm trong khoảng (5, 5) tới (7, 7)</p> <p>Line 2 : Các điểm được giữ lại nằm trong khoảng (7.8, 8) tới (10, 5.25)</p> <p>Line 3 : Loại bỏ</p>
K	2
M	1

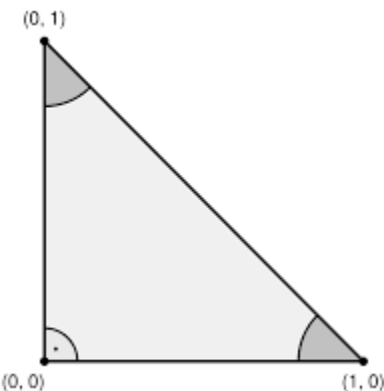
Câu 6 (1 điểm)

H	<p>Cắt xén đoạn thẳng có 2 đầu mút P1 (-1, -2) và P2 (2, 4)</p> <p>Với cửa sổ cắt xén được định nghĩa bởi XL = 0, XR = 1, YB = 0, YT = 1</p>
Đ	<p> $dx = 2 - (-1) = 3$ $dy = 4 - (-2) = 6$ $P1 = -dx = -3$ $q1 = x1 - XL = -1 - 0 = -1$ $q1 / P1 = 1/3$ $P2 = dx = 3$ $q2 = XR - x1 = 1 - (-1) = 2$ $q2 / P2 = 2/3$ $P3 = -dy = -6$ $q3 = y1 - YB = -2$ $q3 / P3 = 1/3$ $P4 = dy = 6$ $q4 = YT - y1 = 3$ $q4 / P4 = 1/2$ for ($P_i < 0$) $t1 = \text{"MAX"} (1/3, 1/3, 0) = 1/3$ for ($P_i > 0$) $t2 = \text{MIN} (2/3, 1/2, 1) = 1/2$ Bởi vì $t1 < t2$ do vậy đây là đoạn hiển thị Tính đầu mút mới $t1 = 1/3$ $x1' = x1 + dx \cdot t1 = -1 + (3 \cdot 1/3) = 0$ $y1' = y1 + dy \cdot t1 = -2 + (6 \cdot 1/3) = 0$ $t2 = 1/2$ $x2' = x1 + dx \cdot t2 = -1 + (3 \cdot 1/2) = 1/2$ </p>

	$y_2' = y_1 + dy \cdot t_2 = -1 + (6 \cdot 1 / 2) = 1$ 
K	2
M	1

III.4. Các phép biến đổi/ đường cong và bề mặt

Câu 1 (1 điểm)

H	<p>a. (0.5 điểm). Giả sử có tam giác vuông đơn vị $\{(0,0), (1,0), (0,1)\}$ như hình dưới. Chúng ta muốn biến đổi nó thành tam giác $\{(1,1), (3,1), (2,2)\}$. Mô tả cách biến đổi và viết ma trận biến đổi tổng hợp.</p>  <p>b. (0.5 điểm) Ta có chuỗi các biến đổi 2 chiều $R(90) T(0,1) R(90) T(2,0)$. Nếu chúng ta có hình vuông đơn vị $\{(0,0), (1,0), (1,1), (0,1)\}$. Vẽ các điểm trung gian trong quá trình biến đổi khi áp dụng liên tiếp, lần lượt các phép biến đổi trên.</p>
Đ	<p>a. Biến đổi tam giác $\{(0,0), (1,0), (0,1)\}$ thành tam giác $\{(1,1), (3,1), (2,2)\}$ bởi phép biến đổi T được thực hiện theo 3 bước sau. Thứ nhất, co giãn bởi $x=2$; Thứ hai: Kéo; Thứ 3; Tịnh tiến với vecto $(1,1)$.</p> <p>b. Xác định 4 điểm của hình vuông. Áp dụng các phép biến đổi vào từng bước</p>
K	2
M	1

Câu 2 (1 điểm)

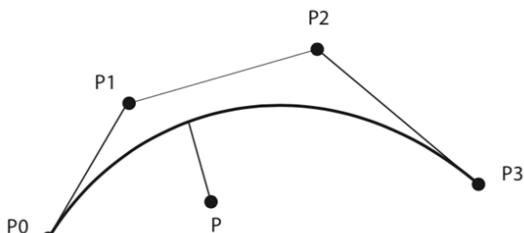
H	a. (0.5 điểm) Thế nào là tọa độ đồng nhất. Cho ví dụ minh họa biến đổi
---	--

	điểm từ tọa độ đồng nhất sang tọa độ Decac vuông góc và ngược lại. b. (0.5 điểm) Mô tả cách biến đổi và viết ma trận biến đổi tổng hợp khi biến đổi tam giác $\{(0,0), (1,0), (0,3)\}$ thành tam giác $\{(-1,0), (-3,0), (-1,-6)\}$. Vẽ tam giác ở mỗi bước.
Đ	a. Thêm thành phần w vào cuối chuỗi các thành phần tọa độ Decac: (x,y) trong 2D thành (x,y,w). Ví dụ (2,3) trong Decac thành (2,3,1). Ngược lại (4,6,2) trong hệ tọa độ đồng nhất thành (2,3) trong Decac. b. Biến đổi tổng hợp được hình thành dựa trên 3 biến đổi riêng rẽ sau: Thứ nhất: quay quanh gốc một góc 1800; Thứ hai, co giãn theo tỉ lệ (2,2); thứ 3, tịnh tiến theo vecto (-1,0).
K	2
M	1

Câu 3 (1 điểm)

H	a. (0.5 điểm) Thế nào là tọa độ đồng nhất. Cho ví dụ minh họa biến đổi điểm từ tọa độ đồng nhất sang tọa độ Decac vuông góc và ngược lại. b. (0.5 điểm) Ba đỉnh của tam giác được biểu diễn bởi A(0, 20), B(40, 40), và C(0, 60). Tịnh tiến tam giác theo chiều x và y lần lượt là 30, 20. Sau đó, quay tam giác đi một góc 30° . Tìm ma trận tổng hợp
Đ	a. Thêm thành phần w vào cuối chuỗi các thành phần tọa độ Decac: (x,y) trong 2D thành (x,y,w). Ví dụ (2,3) trong Decac thành (2,3,1). Ngược lại (4,6,2) trong hệ tọa độ đồng nhất thành (2,3) trong Decac. b. Ma trận tổng hợp: RT
K	2
M	1

Câu 4 (1 điểm)

H	(1 điểm) Xác định đoạn cong Bezier bậc 3 nhận 4 điểm điều khiển P0, P1, P2, và P3.
Đ	$\mathbf{B}(t) = (1-t)^3\mathbf{P}_0 + 3(1-t)^2t\mathbf{P}_1 + 3(1-t)t^2\mathbf{P}_2 + t^3\mathbf{P}_3, t \in [0, 1].$ 

K	2
M	1

Câu 5 (1 điểm)

H	Xác định đường cong Hermite bậc 3
Đ	<p>Gồm 2 bước:</p> <ul style="list-style-type: none"> - đoạn cong Hermite cho từng cặp điểm liên tiếp bằng cách xây dựng ma trận Hermite M sau: $M = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$ <ul style="list-style-type: none"> - ghép các đoạn cong Hermite
K	2
M	1

Câu 6 (1 điểm)

H	<p>a (0.5 điểm). OpenGL chia nhỏ tất cả đa giác (được cho bởi danh sách đỉnh trong cặp lệnh glBegin/glEnd) thành các tam giác trước khi vẽ. Thư viện OpenGL chia đa giác thành tam giác như thế nào? Phương pháp này có cho kết quả tốt khi đa giác là lõm hay không? Tại sao hoặc tại sao không? Cho ví dụ minh họa.</p> <p>b.(0.5 điểm).Thỉnh thoảng một số hệ thống đồ họa dùng hình 4 cạnh thay vì tam giác là hình cơ bản. Nêu lý do tại sao sử dụng tam giác là lựa chọn tốt hơn so với hình bốn cạnh</p>
Đ	<p>a. OpenGL chia nhỏ đa giác thành chuỗi tam giác.</p> <div style="text-align: center;"> </div> <p>Phương pháp này không đưa kết quả tốt trong trường hợp đa giác lõm</p> <p>b. Đối với tam giác, việc nội suy thực hiện dễ dàng hơn vì 3 đỉnh tam giác xác định mặt phẳng. Tam giác được kết xuất đồ họa nhanh hơn hình 4 cạnh.</p>

K	2
M	1

Câu 7 (1 điểm)

H	(1 điểm) Xây dựng đoạn cong Hermite bậc 3
Đ	<p>Đoạn cong Hermite bậc 3 đi qua 2 điểm cho trước cùng 2 tiếp tuyến tại 2 điểm đó</p> <p>Bước 1: Xác định biểu diễn tham số của đường cong bậc 3</p> $X(t) = at^3 + bt^2 + ct + d$ <p>Tính $X'(t)$</p> <p>Bước 2: Dựa vào giá trị điểm và tiếp tuyến cho trước lập 4 phương trình 4 ẩn a, b, c, d tương ứng với $X(0)$, $X(1)$, $X'(0)$ và $X'(1)$</p> <p>Bước 3: tính giá trị a, b, c, d</p> <p>Bước 4: Xác định ma trận đoạn cong Hermite sau</p> $M = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$
K	2
M	1

Câu 8 (1 điểm)

H	(1 điểm) Xây dựng đoạn cong Bezzier bậc 3 theo phương pháp đệ quy
Đ	<p>Xây dựng đệ quy</p> <p>Với 2 điểm P_0 và P_1, đường cong Bézier tuyến tính là một đoạn thẳng nối liền với hai điểm đó.</p> $B(t) = P_0 + t(P_1 - P_0) = (1 - t)P_0 + tP_1, t \in [0, 1]$ <p>Đường cong Bézier bậc 2 được tạo bởi một hàm $B(t)$, với các điểm P_0, P_1, và P_2</p> $B(t) = (1 - t)[(1 - t)P_0 + tP_1] + t[(1 - t)P_1 + tP_2], t \in [0, 1]$
K	2
M	1

Câu 9 (1 điểm)

H	Mô tả các bề mặt mẫu
Đ	Bề mặt trôi: Cho một đường cong $f: [a, b] \rightarrow R^3$ và vectơ $v \in R^3$,

	<p>bề mặt tham số $p: [a,b] \times [0,1] \rightarrow R^3$ được định nghĩa bởi $p(u, t) = f(u) + tv$ được gọi là một bề mặt trời (extrusion). Véc-tơ v được gọi là véc-tơ quét của bề mặt trời. Bề mặt lofted: Cho trước 2 đường cong f và $g: [a, b] \rightarrow R^3$, bề mặt tham số $p: [a,b] \times [0,1] \rightarrow R^3$ được xác định bởi $p(u, v) = (1 - v)f(u) + vg(u)$ (8.3) được gọi là một bề mặt lofted</p>
K	2
M	1


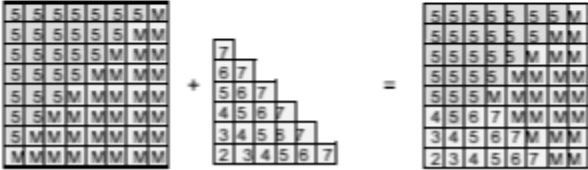
Câu 10 (1 điểm)

H	Mô tả bề mặt song tuyến và bề mặt Coons. So sánh 2 bề mặt này
Đ	<p>Cho điểm p_{00}, p_{01}, p_{10} và p_{11}. Bề mặt song tuyến được định nghĩa như sau: $p(u,v) = (1-v)[(1-u)p_{00} + u.p_{10}] + v[(1-u)p_{01} + u.p_{11}],$ $= (1-u)[(1-v)p_{00} + v.p_{01}] + u[(1-v)p_{10} + v.p_{11}],$ $= (1-u)(1-v)p_{00} + (1-u)v.p_{01} + u(1-v)p_{10} + u.vp_{11}$</p> <p>Bề mặt Coons được định nghĩa như sau: $(P_1p)(u,v) = (1 - u)p(0,v) + up(1,v)$ $(P_2p)(u,v) = (1 - v)p(u,0) + vp(u,1)$ $p(u,v) = (1-v)p(u,0) + vp(u,1) + (1-u)p(0,v) + up(1,v) - (1-u)(1-v)p(0,0) - (1-u)vp(0,1) - u(1-v)p(1,0) - uv p(1,1).$</p> <p>So sánh: Bề mặt song tuyến nội suy giữa hai đường thẳng. Bề mặt Coons dựng các đường cong giữa cặp điểm tương ứng trên hai đường cong</p>
K	2
M	1

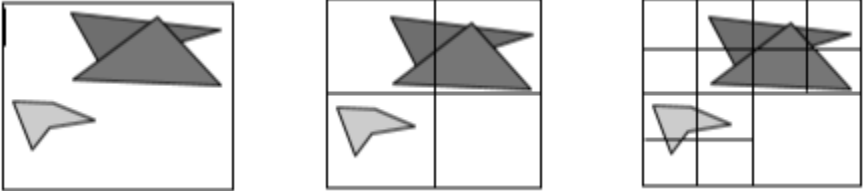
III.5. Các thuật toán xác định bề mặt hiện

Câu 1 (1 điểm)

H	Trình bày thuật toán z-buffer. Cho ví dụ minh họa
Đ	<p>Lưu lại thông tin về độ sâu hiện thời của mỗi điểm Nội suy z trong quá trình tính toán. Lưu trữ một ma trận độ sâu tương ứng với ảnh đầu ra. Mỗi khi xử lý một đa giác, so sánh với các giá trị z đang lưu trữ.</p>

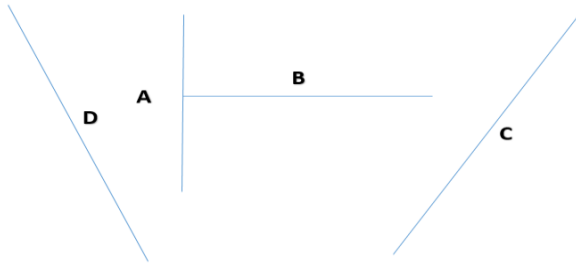
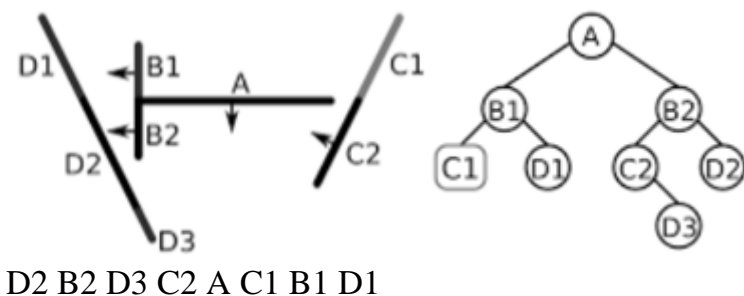
	<p>Lưu lại giá trị màu của những điểm gần nhất. Khởi tạo bộ đệm ảnh với màu nền. Khởi tạo bộ đệm Z với $z = \text{giá trị max. của mặt phẳng clipping.}$ Cần tính giá trị z cho mỗi điểm Bằng cách nội suy từ các đỉnh đa giác. Cập nhật cả bộ đệm ảnh và bộ đệm độ Z. Ví dụ: Vẽ 2 tam giác có độ sâu được mô tả tại từng pixel</p> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div>
K	1
M	1

Câu 2 (1 điểm)

H	(1 điểm) Trình bày ý tưởng thuật toán Warnock. Cho ví dụ minh họa
Đ	<p>Khởi tạo danh sách cửa sổ L (ban đầu: toàn bộ màn hình) Mỗi cửa sổ W trong L tìm cửa sổ thỏa mãn: Tất cả đa giác tách biệt với W: vẽ W với màu nền Chỉ có P giao với W: vẽ phần giao với màu của P còn lại là màu nền. Tìm đa giác bao phủ W và đa giác đó nằm trước tất cả các đa giác khác giao với W: vẽ cửa sổ với màu của đa giác Trường hợp khác: chia cửa sổ thành 4 rồi cho vào L. Lặp lại cho đến khi kích thước cửa sổ là 1 điểm ảnh. Ví dụ:</p> <div style="text-align: center;">  </div>

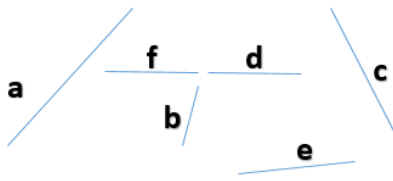
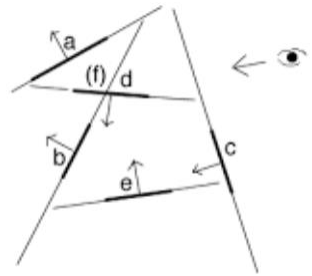
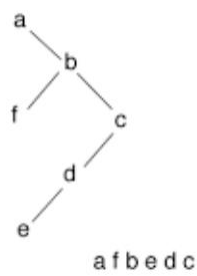
K	1
M	1

Câu 3 (1 điểm)

H	<p>Trình bày ý tưởng thuật toán BSB. Áp dụng cho khung cảnh dưới đây</p> 
Đ	<p><u>Mặt phẳng phân tách</u>: sao cho không có đa giác nào nằm ở nửa không gian chứa điểm nhìn bị một đa giác nằm ở nửa không gian còn lại che khuất</p> <p><u>B1</u>: Dựng cây BSP</p> <p>Chọn đa giác bất kỳ</p> <p>Chia cảnh vật ra 2 nửa không gian: trước và sau.</p> <p>Chia những đa giác nằm ở cả hai nửa không gian thành 2 nhánh</p> <p>Chọn một đa giác ở mỗi nửa – chia đôi cảnh vật tiếp.</p> <p>Tiếp tục chia cho đến khi mỗi phần chỉ còn một đa giác.</p> <p><u>B2</u>: Duyệt cây theo thứ tự InOrder rồi đảo ngược lại danh sách → Thứ tự vẽ</p>  <p>D2 B2 D3 C2 A C1 B1 D1</p>
K	1
M	1

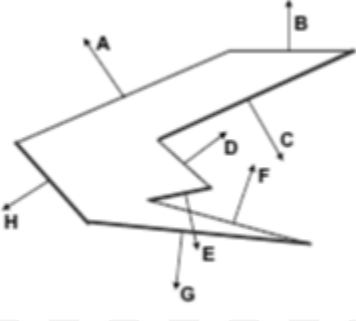
Câu 4 (1 điểm)

H	Trình bày ý tưởng thuật toán BSB. Áp dụng cho khung cảnh dưới đây
---	---

	
Đ	<p><u>Mặt phẳng phân tách</u>: sao cho không có đa giác nào nằm ở nửa không gian chứa điểm nhìn bị một đa giác nằm ở nửa không gian còn lại che khuất</p> <p><u>B1</u>: Dựng cây BSP</p> <p>Chọn đa giác bất kỳ</p> <p>Chia cảnh vật ra 2 nửa không gian: trước và sau.</p> <p>Chia những đa giác nằm ở cả hai nửa không gian thành 2 nhánh</p> <p>Chọn một đa giác ở mỗi nửa – chia đôi cảnh vật tiếp.</p> <p>Tiếp tục chia cho đến khi mỗi phần chỉ còn một đa giác.</p> <p><u>B2</u>: Duyệt cây theo thứ tự InOrder rồi đảo ngược lại danh sách → Thứ tự vẽ</p> <div style="display: flex; justify-content: space-around; align-items: center;">  <div style="text-align: center;">  <p>afbedc</p> </div> </div>
K	1
M	1

Câu 5

H	<p>a. Trình bày giải pháp loại bỏ bề mặt quay vào trong. Cho ví dụ minh họa</p> <p>b. Trình bày lý do đối tượng không hữu hình</p>
Đ	<p>a. V là vectơ hướng nhìn, N là vectơ pháp tuyến của bề mặt</p> <ul style="list-style-type: none"> - $V.N > 0$: Mặt sau - $V.N < 0$: Mặt trước - $V.N = 0$: Song song với hướng nhìn <p>Ví dụ</p>

	 <p>Mặt sau: A, B, D, F Mặt trước: C, E, G, H b. Nằm ngoài vùng hiển thị Quay vào trong (<i>back-facing</i>) Bị che bởi các đối tượng khác gần người quan sát hơn</p>
K	2
M	1

III.6 Ánh sáng/ OpenGL

Câu 1 (1 điểm)

H	(1 điểm) Liệt kê các đối tượng cơ bản (primitives) có thể vẽ được trong OpenGL? Làm thế nào để vẽ được các đối tượng cơ bản trong OpenGL? Cho ví dụ																				
Đ	<table border="1"> <tr> <td>GL_POINT</td><td>Các điểm</td></tr> <tr> <td>GL_LINES</td><td>đoạn thẳng</td></tr> <tr> <td>GL_POLYGON</td><td>Đa giác lõm</td></tr> <tr> <td>GL_TRIANGLES</td><td>Tam giác</td></tr> <tr> <td>GL_QUADS</td><td>Tứ giác</td></tr> <tr> <td>GL_LINE_STRIP</td><td>Đường gấp khúc không khép kín</td></tr> <tr> <td>GL_LINE_LOOP</td><td>Đường gấp khúc khép kín</td></tr> <tr> <td>GL_TRIANGLE_STRIP</td><td>Một giải các tam giác liên kết với nhau</td></tr> <tr> <td>GL_TRIANGLE_FAN</td><td>Các tam giác liên kết với nhau theo hình quạt</td></tr> <tr> <td>GL_QUAD_STRIP</td><td>Một giải các tứ giác liên kết với nhau</td></tr> </table> <p>Các đỉnh được liệt kê trong cặp hàm glBegin(tham_so); ... glEnd(); Ví dụ: glBegin(GL_POLYGON); glVertex2f(0.0, 0.0); glVertex2f(0.0, 3.0); glVertex2f(3.0, 3.0);</p>	GL_POINT	Các điểm	GL_LINES	đoạn thẳng	GL_POLYGON	Đa giác lõm	GL_TRIANGLES	Tam giác	GL_QUADS	Tứ giác	GL_LINE_STRIP	Đường gấp khúc không khép kín	GL_LINE_LOOP	Đường gấp khúc khép kín	GL_TRIANGLE_STRIP	Một giải các tam giác liên kết với nhau	GL_TRIANGLE_FAN	Các tam giác liên kết với nhau theo hình quạt	GL_QUAD_STRIP	Một giải các tứ giác liên kết với nhau
GL_POINT	Các điểm																				
GL_LINES	đoạn thẳng																				
GL_POLYGON	Đa giác lõm																				
GL_TRIANGLES	Tam giác																				
GL_QUADS	Tứ giác																				
GL_LINE_STRIP	Đường gấp khúc không khép kín																				
GL_LINE_LOOP	Đường gấp khúc khép kín																				
GL_TRIANGLE_STRIP	Một giải các tam giác liên kết với nhau																				
GL_TRIANGLE_FAN	Các tam giác liên kết với nhau theo hình quạt																				
GL_QUAD_STRIP	Một giải các tứ giác liên kết với nhau																				

	<pre>glVertex2f(4.0, 1.5); glVertex2f(3.0, 0.0); glEnd();</pre>
K	2
M	1

Câu 2 (1 điểm)

H	(1 điểm) Giải pháp vẽ đường cong trong OpenGL? Cho ví dụ minh họa
Đ	<p>Từ phương trình đường cong, lấy mẫu. Từ đó nối từng cặp điểm liên tiếp ở dạng đường thẳng</p> <p>Ví dụ:</p> <p>Với đường cong Rose</p> <pre>glBegin(GL_LINE_STRIP); for(int i = 0; i <= 180; i++){ glPushMatrix(); float temp = radius*sin(limit*i*radian); twodimpoint point = {temp*cos(i*radian), temp*sin(i*radian)}; glVertex2fv(point); glPopMatrix(); } glEnd();</pre>
K	2
M	1

Câu 3 (1 điểm)

H	(1 điểm) Nêu giải pháp thực hiện xoắn (tweening) giữa 2 đa giác
Đ	<pre>void drawTweening(float** A, float** B, int n, float t) { glBegin(GL_LINE_LOOP); for (int i = 0; i < n; ++i) { float Px = tweening(A[i][0], B[i][0], t); float Py = tweening(A[i][1], B[i][1], t); glVertex2f(Px, Py); } glEnd(); glFlush(); } void display(void) { for (float t = 0.0, deltaT = 0.01; ; t += deltaT) { glClear(GL_COLOR_BUFFER_BIT);</pre>

	<pre> drawTweening(A, B, numberOfVertexes, t); glutSwapBuffers(); if (t >= 1.0 t < 0.0) deltaT = -deltaT; Sleep(100); } } </pre>
K	2
M	1

Câu 4 (1 điểm)

H	(1 điểm) a. Thế nào là mô hình ánh sáng? Phân loại mô hình ánh sáng? Phân biệt mô hình ánh sáng và mô hình tạo bóng?
Đ	<p>Mô hình ánh sáng: Các luật đơn giản về tương tác giữa vật thể và ánh sáng</p> <p>Hai thành phần quan trọng: tính chất bề mặt và tính chất ánh sáng</p> <p>Mô hình ánh sáng được chia làm 2 loại: cục bộ và toàn cục</p> <p>Mô hình tạo bóng: Thiết lập màu sắc và cường độ sáng tại tất cả các điểm trên bề mặt. Toàn diện hơn mô hình ánh sáng</p>
K	2
M	1

Câu 5 (1 điểm)

H	(1 điểm) Liệt kê ít nhất 2 mô hình ánh sáng? Mô tả đặc điểm của từng loại
Đ	<p>Bouknight:</p> $I_{\lambda} = I_a(\lambda)k_a(\lambda) + I_p(\lambda)k_d(\lambda)r_d$ <p>Phong:</p> $I_{\lambda} = I_a k_a + I_p [k_d \cos \theta + k_s \cos^n \alpha]$
K	2
M	1

Câu 6 (1 điểm)

H	(1 điểm) Liệt kê ít nhất 3 mô hình tạo bóng? Mô tả đặc điểm của từng loại
Đ	<p>Tạo bóng phẳng (Flat /facet shading)</p> <p>Mỗi đa giác là một màu</p> <p>Tạo bóng nội suy (Interpolated shading)</p> <p>Nội suy màu ở vùng cạnh</p> <p>Tạo bóng Gouraud</p> <p>Tìm vec-tơ pháp tuyến cho mỗi đỉnh bằng cách lấy trung bình các pháp tuyến bề mặt, hoặc thông qua phân tích.</p> <p>Sử dụng pháp tuyến với mô hình tạo bóng nào đó,</p> <p>Nội suy cường độ màu sắc của đỉnh dọc theo các cạnh.</p>

	Nội suy giá trị các cạnh theo đường quét.
K	2
M	1

IV. Ma trận đề thi

Đề thi: Môn đồ họa máy tính

Thời gian: 60 phút

Trắc nghiệm: 20 câu (5 điểm)

Tự luận: 5 câu (5 điểm)

Số TT	Kho	Trắc nghiệm	Tự luận
1	Thiết bị	2	
2	Các thuật toán tô màu	2	1
3	Các thuật toán vẽ đường thẳng, đường tròn	2	1
4	Các thuật toán cắt xén	2	1
5	Các phép biến đổi	3	1
6	Đường cong và bề mặt	2	
7	Xác định bề mặt hiện	3	1
8	Ánh sáng	2	
9	OpenGL	2	
Tổng số		20	5