Short Questions and Answers

1. What are the different application areas of computer graphics?

Computer graphics are used in various fields including entertainment (movies, video games), design (CAD, CAM), education (simulations, visualizations), and scientific research (data visualization, modeling). They are essential for creating visual content, enhancing user interfaces, and providing immersive experiences.

2. What components constitute a graphics system?

A graphics system typically includes hardware components like a graphics card (GPU), monitor, and input devices (mouse, keyboard). Software components include graphics libraries (OpenGL, DirectX), graphics drivers, and applications that generate and manipulate graphical content.

3. What are video-display devices?

Video-display devices are hardware used to display visual output from a computer. Examples include CRT monitors, LCD screens, LED displays, and projectors. They convert digital signals from the graphics system into visual images.

4. Describe the difference between raster-scan systems and random-scan systems.

Raster-scan systems display images by scanning lines from top to bottom and updating pixels row by row. Random-scan systems, also known as vector displays, draw images directly by tracing lines and shapes. Raster systems are better for complex images, while random systems excel at line art.

5. What are graphics monitors and workstations?

Graphics monitors are high-resolution screens designed for detailed visual output. Graphics workstations are powerful computers equipped with advanced graphics capabilities and hardware to handle intensive graphical tasks, such as 3D modeling and animation.

6. What are input devices in computer graphics?

Input devices for computer graphics include the mouse, keyboard, graphics tablets, joysticks, and 3D mice. These devices allow users to interact with and manipulate graphical content in applications.

7. What are output primitives in computer graphics?

Output primitives are the basic elements used to create images in computer graphics. They include points, lines, curves, and polygons. These primitives are combined and manipulated to produce complex graphical scenes.

8. What is the Digital Differential Analyzer (DDA) used for?

The Digital Differential Analyzer (DDA) is an algorithm used for interpolating values over an interval, often used in graphics to draw lines and curves. It incrementally plots points along a path to create smooth lines.

9. Explain Bresenham's Algorithm.

Bresenham's Algorithm is an efficient way to draw straight lines on a raster grid. It uses integer arithmetic to determine the closest pixel to the theoretical line, reducing computational overhead compared to floating-point calculations.

10. What are circle-generating algorithms?

Circle-generating algorithms, such as the Midpoint Circle Algorithm, plot points of a circle by calculating the positions of pixels that approximate the circle's outline. These algorithms ensure symmetry and minimize computational cost.

11. Describe the scan-line algorithm for polygon filling.

The scan-line algorithm fills polygons by processing each horizontal line (scan-line) of the frame buffer. It calculates intersections of the scan-line with polygon edges and fills the pixels between pairs of intersections.

12. What are boundary-fill and flood-fill algorithms?

Boundary-fill and flood-fill algorithms are used to fill areas with color. Boundary-fill fills the area within a boundary color, while flood-fill fills an area starting from a seed point and spreading outwards until it hits a boundary.

13. What are the basic 2-D geometric transformations?

Basic 2-D geometric transformations include translation (moving objects), scaling (resizing objects), rotation (turning objects), reflection (mirroring objects), and shearing (skewing objects). These transformations manipulate the position and orientation of graphical objects.

14. Explain translation transformation.

Translation transformation moves an object from one location to another by adding a specified distance to the coordinates of its points. This is represented by a translation matrix.

15. What is scaling transformation?

Scaling transformation changes the size of an object by multiplying the coordinates of its points by scale factors along the x and y axes. Uniform scaling uses the same factor for both axes, while non-uniform scaling uses different factors.

16. Describe rotation transformation.

Rotation transformation turns an object around a specified pivot point by a certain angle. The coordinates of the object's points are recalculated using trigonometric functions to achieve the rotation.

17. What is reflection transformation?

Reflection transformation creates a mirror image of an object relative to a specified axis (x-axis, y-axis, or an arbitrary line). It is useful for creating symmetrical designs.

18. Explain shear transformation.

Shear transformation skews the shape of an object along the x or y axis by shifting its points in a direction parallel to the axis. It changes the shape but not the area of the object.

19. How are transformations represented using matrices?

Transformations are represented using matrices to enable efficient computation and combination of multiple transformations. A transformation matrix is applied to the coordinates of an object's points to perform the desired transformation.

20. What are homogeneous coordinates?

Homogeneous coordinates introduce an extra dimension to standard Cartesian coordinates, allowing translation to be represented as a matrix multiplication. This simplifies the combination of transformations into a single matrix operation.

21. What are composite transforms?

Composite transforms combine multiple basic transformations (translation, scaling, rotation, etc.) into a single transformation matrix. This allows for the application of several transformations in one step.

22. How do transformations between coordinate systems work?

Transformations between coordinate systems involve converting coordinates from one frame of reference to another, often using transformation matrices. This is essential in rendering scenes from different viewpoints or aligning objects in 3D space.

23. What is the viewing pipeline in computer graphics?

The viewing pipeline is a sequence of steps that transforms 3D objects into a 2D screen representation. It includes modeling, viewing, clipping, and projection transformations, ensuring correct display of the 3D scene.

24. Explain the concept of point clipping.

Point clipping determines whether a point lies within a specified rectangular viewport. Points outside the viewport are discarded, ensuring only visible points are rendered on the screen.

25. What is the Cohen-Sutherland line clipping algorithm?

The Cohen-Sutherland algorithm clips lines to a rectangular viewport using a divide-and-conquer approach. It assigns region codes to endpoints, determines line visibility, and clips or discards segments based on these codes.

26. Describe polygon clipping.

Polygon clipping involves cutting a polygon to fit within a rectangular viewport. Algorithms like Sutherland-Hodgman and Weiler-Atherton handle the clipping, ensuring that the resulting polygon remains within the visible area.

27. What are the methods for 3-D object representation?

3-D objects can be represented using various methods such as polygonal meshes, quadric surfaces, parametric curves and surfaces, and implicit surfaces. Each method has its advantages for different types of objects and applications.

28. What are polygon surfaces?

Polygon surfaces use a mesh of polygons (usually triangles or quadrilaterals) to represent 3-D objects. They are widely used due to their simplicity and efficiency in rendering complex shapes.

29. What are quadric surfaces?

Quadric surfaces are defined by second-degree polynomial equations in three variables (x, y, z). Examples include spheres, ellipsoids, cylinders, and paraboloids. They are used for smooth, mathematically defined shapes.

30. What is spline representation?

Spline representation uses piecewise polynomial functions to define curves and surfaces. Splines provide smooth and flexible modeling of shapes, making them suitable for complex, organic forms in graphics.

31. Explain the characteristics of Hermite curves.

Hermite curves are defined by endpoints and tangent vectors at those points. They provide local control over the shape of the curve, allowing precise adjustments to the curve's direction and curvature.

32. What are Bezier curves?

Bezier curves are defined by control points, with the curve shape influenced by these points. They offer intuitive control and are widely used in graphics and design for creating smooth, scalable curves.

33. Describe B-Spline curves.

B-Spline curves are defined by a set of control points and basis functions. They provide greater flexibility and local control compared to Bezier curves, making them suitable for complex curve modeling.

34. What is the significance of homogeneous coordinates?

Homogeneous coordinates simplify the mathematical representation of geometric transformations, especially translation. They enable the use of matrix multiplication for all transformations, facilitating the combination and inversion of transformations.

35. How does the midpoint circle algorithm work?

The midpoint circle algorithm efficiently draws circles by determining the next pixel position based on the previous one. It uses integer arithmetic to minimize calculations, ensuring symmetry and reducing computational overhead.

36. What are translation transformations?

Translation transformations move an object by adding a specified distance to the coordinates of all its points. This changes the object's position without altering its shape or orientation.

37. Describe scaling transformations.

Scaling transformations resize an object by multiplying the coordinates of its points by scale factors. This can enlarge or shrink the object uniformly or non-uniformly depending on the scale factors for each axis.

38. What are rotation transformations?

Rotation transformations turn an object around a specified pivot point by a certain angle. The new coordinates are calculated using trigonometric functions based on the angle of rotation.

39. What is reflection transformation used for?

Reflection transformation creates a mirror image of an object relative to a specified axis. It is used for symmetrical designs and to create reflections of objects in scenes.

40. Explain shear transformation.

Shear transformation skews the shape of an object along one axis by shifting points parallel to another axis. This distorts the object's shape without changing its area.

41. What are composite transforms?

Composite transforms combine multiple transformations (like translation, scaling, and rotation) into a single matrix operation. This allows for complex transformations to be applied in one step.

42. How do transformations between coordinate systems facilitate rendering?

Transformations between coordinate systems convert object coordinates to world coordinates, then to camera coordinates, and finally to screen coordinates. This ensures accurate placement and viewing of objects in a rendered scene.

43. What is the viewing pipeline?

The viewing pipeline includes steps that transform 3D objects to 2D screen representation. It involves modeling transformations, viewing transformations, clipping, and projection, ensuring the correct display of the scene.

44. What is point clipping?

Point clipping involves checking if a point lies within a defined viewport. Points outside this area are not displayed, ensuring only visible points are rendered.

45. Explain the Cohen-Sutherland line clipping algorithm.

The Cohen-Sutherland algorithm clips lines to a viewport using region codes for endpoints. It determines visibility and clips or discards segments based on these codes, efficiently handling line visibility.

46. Describe polygon clipping.

Polygon clipping cuts polygons to fit within a viewport. Algorithms like Sutherland-Hodgman handle this by iterating through polygon vertices, ensuring the clipped polygon remains within the visible area.

47. What are the different methods for 3-D object representation?

3-D objects can be represented using polygonal meshes, quadric surfaces, parametric curves, and implicit surfaces. These methods offer various advantages for modeling different types of objects.

48. How are quadric surfaces represented?

Quadric surfaces are represented by second-degree polynomial equations. Examples include spheres and cylinders. They provide a smooth, mathematically defined representation for certain types of shapes.

49. What is the significance of spline representation?

Spline representation offers smooth and flexible modeling of curves and surfaces. It is widely used for complex, organic forms in graphics due to its ability to provide local control and smooth transitions.

50. Describe the characteristics of Hermite curves.

Hermite curves are defined by endpoints and tangents, offering local control over shape. This allows precise adjustments to the curve's direction and curvature, useful in animation and modeling.

51. What are Bezier curves?

Bezier curves are defined by control points and offer intuitive control over curve shape. They are commonly used in design and graphics for creating smooth, scalable curves.

52. What are B-Spline curves?

B-Spline curves are defined by control points and basis functions, providing greater flexibility and local control than Bezier curves. They are used for modeling complex shapes and curves.

53. What are homogeneous coordinates?

Homogeneous coordinates add an extra dimension to Cartesian coordinates, allowing translation to be represented as matrix multiplication. This simplifies the combination of transformations.

54. How does the midpoint circle algorithm efficiently draw circles?

The midpoint circle algorithm plots circles by determining the next pixel based on the previous one, using integer arithmetic for efficiency. It ensures symmetry and reduces computational effort.

55. What are translation transformations?

Translation transformations move an object by adding a specified distance to all its points' coordinates. This changes the object's position without altering its shape or orientation.

56. What are scaling transformations?

Scaling transformations resize an object by multiplying its points' coordinates by scale factors. This can uniformly or non-uniformly enlarge or shrink the object.

57. What are rotation transformations?

Rotation transformations turn an object around a pivot point by a specific angle, recalculating coordinates using trigonometric functions to achieve the rotation.

58. What is reflection transformation?

Reflection transformation creates a mirror image of an object relative to a specified axis, useful for symmetrical designs and reflections in scenes.

59. Explain shear transformation.

Shear transformation skews an object's shape along one axis by shifting points parallel to another axis, distorting the shape without changing its area.

60. How are transformations represented using matrices?

Transformations are represented using matrices for efficient computation and combination. A transformation matrix applied to object coordinates performs the desired transformation.

61. What are homogeneous coordinates?

Homogeneous coordinates add an extra dimension to Cartesian coordinates, simplifying translation representation and enabling matrix multiplication for transformations.

62. What are composite transforms?

Composite transforms combine multiple transformations into a single matrix, allowing complex transformations to be applied in one operation.

63. How do transformations between coordinate systems facilitate rendering?

Transformations between coordinate systems convert coordinates through stages (object to world, camera, screen) ensuring accurate object placement and viewing.

64. What is the viewing pipeline?

The viewing pipeline includes modeling, viewing, clipping, and projection transformations to convert 3D objects into a 2D screen representation, ensuring correct display.

65. What is point clipping?

Point clipping checks if a point lies within a viewport, discarding points outside this area to ensure only visible points are rendered.

66. What is the Cohen-Sutherland line clipping algorithm?

The Cohen-Sutherland algorithm clips lines using region codes for endpoints, determining visibility and clipping segments based on these codes for efficient handling.

67. Describe polygon clipping.

Polygon clipping involves cutting polygons to fit within a viewport, using algorithms like Sutherland-Hodgman to ensure the clipped polygon remains visible.

68. What are the different methods for 3-D object representation?

Methods for 3-D object representation include polygonal meshes, quadric surfaces, parametric curves, and implicit surfaces, each offering advantages for different modeling needs.

69. How are quadric surfaces represented?

Quadric surfaces are defined by second-degree polynomial equations, representing smooth shapes like spheres and cylinders mathematically.

70. What is the significance of spline representation?

Spline representation provides smooth, flexible modeling of curves and surfaces, with local control and smooth transitions, ideal for complex forms in graphics.

71. Describe the characteristics of Hermite curves.

Hermite curves are defined by endpoints and tangent vectors, offering local control over shape for precise adjustments in animation and modeling.

72. What are Bezier curves?

Bezier curves are defined by control points, providing intuitive control over curve shape, widely used in design for creating smooth, scalable curves.

73. What are B-Spline curves?

B-Spline curves use control points and basis functions for greater flexibility and local control than Bezier curves, suitable for complex shapes and curves.

74. What are homogeneous coordinates?

Homogeneous coordinates add an extra dimension to Cartesian coordinates, simplifying translation representation and enabling matrix multiplication for transformations.

75. How does the midpoint circle algorithm efficiently draw circles?

The midpoint circle algorithm plots circles using integer arithmetic to determine the next pixel, ensuring symmetry and reducing computational effort.

76. What are translation transformations?

Translation transformations move an object by adding a specified distance to all its points' coordinates, changing the object's position without altering its shape.

77. What are scaling transformations?

Scaling transformations resize an object by multiplying its points' coordinates by scale factors, uniformly or non-uniformly enlarging or shrinking the object.

78. What are rotation transformations?

Rotation transformations turn an object around a pivot point by a specific angle, using trigonometric functions to recalculate coordinates for the rotation.

79. What is reflection transformation?

Reflection transformation creates a mirror image of an object relative to a specified axis, used for symmetrical designs and reflections in scenes.

80. Explain shear transformation.

Shear transformation skews an object's shape along one axis by shifting points parallel to another axis, distorting the shape without changing its area.

81. What are composite transforms?

Composite transforms combine multiple transformations into a single matrix operation, allowing complex transformations to be applied in one step.

82. How do transformations between coordinate systems facilitate rendering?

Transformations convert coordinates through stages (object to world, camera, screen), ensuring accurate placement and viewing in the rendered scene.

83. What is the viewing pipeline?

The viewing pipeline transforms 3D objects to 2D screen representation through modeling, viewing, clipping, and projection transformations, ensuring correct display.

84. What is point clipping?

Point clipping checks if a point lies within a viewport, discarding points outside the area to ensure only visible points are rendered.

85. What is the Cohen-Sutherland line clipping algorithm?

The Cohen-Sutherland algorithm clips lines using region codes for endpoints, efficiently handling line visibility and clipping segments based on these codes.

86. Describe polygon clipping.

Polygon clipping cuts polygons to fit within a viewport, using algorithms like Sutherland-Hodgman to ensure the clipped polygon remains visible.

87. What are the different methods for 3-D object representation?

Methods include polygonal meshes, quadric surfaces, parametric curves, and implicit surfaces, each offering advantages for different types of 3-D modeling.

88. How are quadric surfaces represented?

Quadric surfaces are defined by second-degree polynomial equations, representing smooth, mathematically defined shapes like spheres and cylinders.

89. What is the significance of spline representation?

Spline representation provides smooth, flexible modeling of curves and surfaces, with local control and smooth transitions, ideal for complex forms.

90. Describe the characteristics of Hermite curves.

Hermite curves are defined by endpoints and tangents, offering local control over shape, allowing precise adjustments in animation and modeling.

91. What are Bezier curves?

Bezier curves are defined by control points, providing intuitive control over curve shape, widely used in design for creating smooth, scalable curves.

92. What are B-Spline curves?

B-Spline curves use control points and basis functions, offering greater flexibility and local control than Bezier curves, suitable for complex shapes.

93. What are homogeneous coordinates?

Homogeneous coordinates add an extra dimension to Cartesian coordinates, simplifying translation representation and enabling matrix multiplication for transformations.

94. How does the midpoint circle algorithm efficiently draw circles?

The midpoint circle algorithm plots circles using integer arithmetic to determine the next pixel, ensuring symmetry and reducing computational effort.

95. What are translation transformations?

Translation transformations move an object by adding a specified distance to all its points' coordinates, changing the object's position without altering its shape.

96. What are scaling transformations?

Scaling transformations resize an object by multiplying its points' coordinates by scale factors, uniformly or non-uniformly enlarging or shrinking the object.

97. What are rotation transformations?

Rotation transformations turn an object around a pivot point by a specific angle, using trigonometric functions to recalculate coordinates for the rotation.

98. What is reflection transformation?

Reflection transformation creates a mirror image of an object relative to a specified axis, used for symmetrical designs and reflections in scenes.

99. Explain shear transformation.

Shear transformation skews an object's shape along one axis by shifting points parallel to another axis, distorting the shape without changing its area.

100. What are composite transforms?

Composite transforms combine multiple transformations into a single matrix, allowing complex transformations to be applied in one operation.

101. How do transformations between coordinate systems facilitate rendering?

Transformations convert coordinates through stages (object to world, camera, screen), ensuring accurate placement and viewing in the rendered scene.

102. What is the viewing pipeline?

The viewing pipeline transforms 3D objects to 2D screen representation through modeling, viewing, clipping, and projection transformations, ensuring correct display.

103. What is point clipping?

Point clipping checks if a point lies within a viewport, discarding points outside the area to ensure only visible points are rendered.

104. What is the Cohen-Sutherland line clipping algorithm?

The Cohen-Sutherland algorithm clips lines using region codes for endpoints, efficiently handling line visibility and clipping segments based on these codes.

105. Describe polygon clipping.

Polygon clipping cuts polygons to fit within a viewport, using algorithms like Sutherland-Hodgman to ensure the clipped polygon remains visible.

106. What are the different methods for 3-D object representation?

Methods include polygonal meshes, quadric surfaces, parametric curves, and implicit surfaces, each offering advantages for different types of 3-D modeling.

107. How are quadric surfaces represented?

Quadric surfaces are defined by second-degree polynomial equations, representing smooth, mathematically defined shapes like spheres and cylinders.

108. What is the significance of spline representation?

Spline representation provides smooth, flexible modeling of curves and surfaces, with local control and smooth transitions, ideal for complex forms.

109. Describe the characteristics of Hermite curves.

Hermite curves are defined by endpoints and tangents, offering local control over shape, allowing precise adjustments in animation and modeling.

110. What are Bezier curves?

Bezier curves are defined by control points, providing intuitive control over curve shape, widely used in design for creating smooth, scalable curves.

111. What are B-Spline curves?

B-Spline curves use control points and basis functions, offering greater flexibility and local control than Bezier curves, suitable for complex shapes.

112. What are homogeneous coordinates?

Homogeneous coordinates add an extra dimension to Cartesian coordinates, simplifying translation representation and enabling matrix multiplication for transformations.

113. How does the midpoint circle algorithm efficiently draw circles?

The midpoint circle algorithm plots circles using integer arithmetic to determine the next pixel, ensuring symmetry and reducing computational effort.

114. What are translation transformations?

Translation transformations move an object by adding a specified distance to all its points' coordinates, changing the object's position without altering its shape.

115. What are scaling transformations?

Scaling transformations resize an object by multiplying its points' coordinates by scale factors, uniformly or non-uniformly enlarging or shrinking the object.

116. What are rotation transformations?

Rotation transformations turn an object around a pivot point by a specific angle, using trigonometric functions to recalculate coordinates for the rotation.

117. What is reflection transformation?

Reflection transformation creates a mirror image of an object relative to a specified axis, used for symmetrical designs and reflections in scenes.

118. Explain shear transformation.

Shear transformation skews an object's shape along one axis by shifting points parallel to another axis, distorting the shape without changing its area.

119. What are composite transforms?

Composite transforms combine multiple transformations into a single matrix operation, allowing complex transformations to be applied in one step.

120. How do transformations between coordinate systems facilitate rendering?

Transformations convert coordinates through stages (object to world, camera, screen), ensuring accurate placement and viewing in the rendered scene.

121. What is the viewing pipeline?

The viewing pipeline transforms 3D objects to 2D screen representation through modeling, viewing, clipping, and projection transformations, ensuring correct display.

122. What is point clipping?

Point clipping checks if a point lies within a viewport, discarding points outside the area to ensure only visible points are rendered.

123. What is the Cohen-Sutherland line clipping algorithm?

The Cohen-Sutherland algorithm clips lines using region codes for endpoints, efficiently handling line visibility and clipping segments based on these codes.

124. Describe polygon clipping.

Polygon clipping cuts polygons to fit within a viewport, using algorithms like Sutherland-Hodgman to ensure the clipped polygon remains visible.

125. What is the significance of polygon clipping in computer graphics?

Polygon clipping ensures that only the visible parts of a polygon are rendered within the viewport. This enhances performance and visual accuracy by eliminating unnecessary drawing outside the viewable area.