

## 1 Построение изображений двумерных кривых

Написать программу, строящую изображение заданной замечательной кривой.

$$1. \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1.$$

$$2. (x^2 + y^2)^2 = a^2(x^2 - y^2).$$

$$3. x = a \cos^3 \varphi, y = a \sin^3 \varphi.$$

$$4. \rho = a^2 \cos 2\varphi.$$

$$5. x^{2/3} + y^{2/3} = a^{2/3}.$$

$$6. (x^2 + y^2 + ax)^2 = a^2(x^2 + y^2).$$

$$7. x = a \sin t, y = b \cos t.$$

$$8. y = ax^{3/2}, 0 \leq x \leq B.$$

$$9. \rho = a\varphi, 0 \leq \varphi \leq B.$$

$$10. \rho = a/\varphi, 0 < A \leq \varphi \leq B.$$

$$11. \rho = a \cos(3\varphi).$$

$$12. \rho = a \sin(2\varphi).$$

$$13. \rho = a(1 - \cos \varphi).$$

$$14. \rho = a \cos(2\varphi) / \cos \varphi, \\ -\pi < A \leq \varphi \leq B < a.$$

$$15. x = a\varphi - b \sin \varphi, y = a - b \cos \varphi, \\ a < b, A \leq \varphi \leq B.$$

$$16. x = 3at/(1 + t^3), y = 3at^2/(1 + t^3), \\ -1 < A \leq t \leq B.$$

$$17. x = 3at/(1 + t^3), y = 3at^2/(1 + t^3), \\ A \leq t \leq B < -1.$$

$$18. y^2 = x^3/(a - x), 0 < x \leq B.$$

$$19. \rho = a \cos(7\varphi).$$

$$20. \rho = a \sin(6\varphi).$$

$x, y$  — декартовы координаты,  $\rho, \varphi$  — полярные,  $t$  — независимый параметр.  $a, b, k, A, B$  — константы, значения которых выбираются из соображений наглядности.  $a, b > 0$ .

## 2 Каркасная визуализация выпуклого многоугольника. Удаление невидимых линий

Разработать формат представления многогранника и процедуру его каркасной отрисовки в ортографической и изометрической проекциях. Обеспечить удаление невидимых линий и возможность пространственных поворотов и масштабирования многогранника. Обеспечить автоматическое центрирование и изменение размеров изображения при изменении размеров окна.

1. Куб.
2. Правильный октаэдр.
3. Параллелепипед.
4. Клин.
5. Обелиск (усеченный клин).
6. Усеченный правильный тетраэдр (грани — правильные треугольники и шестиугольники).
7. 4-гранная прямая правильная призма.
8. 5-гранная прямая правильная призма.
9. 6-гранная прямая правильная призма.
10. 8-гранная прямая правильная призма.
11. 10-гранная прямая правильная призма.
12. 5-гранная прямая правильная пирамида.
13. 6-гранная прямая правильная пирамида.
14. 8-гранная прямая правильная пирамида.
15. 12-гранная прямая правильная пирамида.
16. 16-гранная прямая правильная пирамида.
17. 4-гранная прямая правильная усеченная пирамида.
18. 5-гранная прямая правильная усеченная пирамида.
19. 6-гранная прямая правильная усеченная пирамида.
20. 8-гранная прямая правильная усеченная пирамида.
21. 10-гранная прямая правильная усеченная пирамида.
22. Правильный додекаэдр.
23. Правильный икосаэдр.

### **3 Основы построения фотореалистичных изображений**

Используя результаты л. р. №2, аппроксимировать заданное тело выпуклым многогранником. Точность аппроксимации задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель закраски для случая одного источника света. Параметры освещения и отражающие свойства материала задаются пользователем.

1. Прямой усеченный эллиптический конус.
2. Прямой эллиптический цилиндр.
3. Шар.
4. Полушарие.
5. Эллипсоид.
6. Параболоид.
7. Одна из полостей двуполостного гиперболоида.
8. Наклонный круговой цилиндр.
9. Шаровой слой.
10. Шаровой сектор.
11. Прямой усеченный круговой конус.
12. Прямой круговой цилиндр.
13. Усеченный прямой круговой цилиндр.
14. Усеченный прямой эллиптический цилиндр.
15. Сектор эллипсоида.
16. Прямой цилиндр, основание — сектор параболы.
17. Прямой цилиндр, основание — сектор гиперболы.
18. Слой параболоида.
19. Слой полости двуполостного гиперболоида.
20. Слой эллипсоида.
21. Цилиндрическая подкова.
22. Бочка.

#### 4 Ознакомление с библиотеками трехмерной графики. Создание простой программы с трехмерной графикой

Используя результаты л. р. №3, изобразить заданное тело (то же, что и в л. р. №3) с использованием средств библиотеки трехмерной графики (предпочтительно OpenGL) без применения готовых структур для хранения тел. Точность аппроксимации задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей.

Параметры освещения и отражающие свойства материала задаются пользователем.

#### 5 Построение плоских полиномиальных кривых

Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиций точек и, при необходимости, значений касательных векторов и натяжения.

1. Интерполяционный многочлен Лагранжа по пяти точкам.
2. Сегмент кубического сплайна по конечным точкам и касательным.
3. Сплайн непрерывной кривизны из двух сегментов по трем точкам и касательным в 1-ой и 3-ей точках.
4. Кривая Безье 3-ей степени.
5. Кривая Безье 2-ой степени.
6. Кривая Безье 4-ой степени.
7. Кривая Безье 5-ой степени.
8. Сегмент кривой Кэтмулла-Рома (Catmull-Rom).
9. Фундаментальная кривая (cardinal spline) 3-ей степени из двух сегментов. Предусмотреть изменение натяжений.
10. В-сплайн,  $n = 6$ ,  $k = 3$ , узловой вектор равномерный.
11. В-сплайн,  $n = 5$ ,  $k = 3$ , узловой вектор равномерный.
12. В-сплайн,  $n = 6$ ,  $k = 4$ , узловой вектор равномерный.
13. Интерполяционный многочлен Лагранжа по четырем точкам.
14. Интерполяционный многочлен Лагранжа по шести точкам.
15. NURBS,  $n = 6$ ,  $k = 3$ , узловой вектор равномерный, веса точек различны и изменяемы.
16. NURBS,  $n = 6$ ,  $k = 3$ , узловой вектор неравномерный, веса точек различны и изменяемы.
17. NURBS,  $n = 5$ ,  $k = 3$ , узловой вектор неравномерный, веса точек различны и изменяемы.
18. Два сопряженных сегмента кривой Безье 3-ей степени по шести точкам (внутренняя точка доопределяется).
19. В-сплайн,  $n = 5$ ,  $k = 4$ , узловой вектор равномерный.
20. NURBS,  $n = 5$ ,  $k = 4$ , узловой вектор равномерный, веса точек различны и изменяемы.

## 6 Создание шейдерных эффектов

Программно сгенерировать заданную поверхность, наложить на нее подходящую по размеру и пропорциям текстуру и применить указанный эффект. Обеспечить просмотр объекта с разных точек, динамическую загрузку текстуры из файла, включение и отключение шейдерного эффекта, регулирование скорости анимации эффекта, если это необходимо.

	Поверхность	Эффект
1	Геликоид: $\begin{cases} x = u \cos v, \\ y = u \sin v, \\ z = v \end{cases}$	Анимация. Координата $X$ изменяется по закону $X = X \cos t$
2	Лист Мебиуса: $\begin{cases} x = (1 + \frac{v}{2} \cos \frac{u}{2}) \cos u, \\ y = (1 + \frac{v}{2} \cos \frac{u}{2}) \sin u, \\ z = \frac{v}{2} \sin \frac{u}{2}, \\ 0 \leq u \leq 2\pi, -1 \leq v \leq 1 \end{cases}$	Анимация. Цветовые координаты изменяются по синусоидальному закону
3	$z = x^2 + y^2, x, y \in [-1; 1]$	Зеркальное освещение от источника света в заданной позиции
4	$z = x + y^2, x, y \in [-1; 1]$	Анимация. Вращение относительно оси $OY$
5	$z = y^2 \sin x,$ $x \in [-\pi; \pi], y \in [-1; 1]$	Анимация. Координата $X$ изменяется по закону $X \cos t$ , координата $Y$ изменяется по закону $Y = Y \sin(X + t)$
6	$z = \sin x + \cos y, x, y \in [-\pi; \pi]$	Получение изометрической проекции сцены
7	$z = \operatorname{sh} x \operatorname{ch} y, x, y \in [-3; 3]$	Анимация. Координата $X$ изменяется по закону $X = \sin t$ для всех вершин, компонента $X$ нормали которых больше 0
8	$x = \cos u, y = v, z = uv,$ $u \in [0; 2\pi], v \in [-1; 1]$	Прозрачность вершины обратно пропорциональна расстоянию до заданной точки
9	$x = u^2, y = v^2, z = u + v,$ $u, v \in [-3; 3]$	Прозрачность вершины пропорциональна косинусу угла между нормалью и направлением на заданную точку
10	$z = x^2 - y^2, x, y \in [-1; 1]$	Зеркальное освещение от источника света, расположенного в заданной позиции
11	$z = 2x^2 + y^2, x, y \in [-1; 1]$	Анимация. Изменение интенсивности источника рассеянного света по синусоидальному закону

12	$z = y^2 - x, x, y \in [-1; 1]$	Анимация. Координата $Y$ изменяется по закону $Y = Y \cos(t + Y)$
13	$z = y^2 \sin x + x^2,$ $x \in [-\pi; \pi], y \in [-1; 1]$	Анимация. Прозрачность изменяется по синусоидальному закону
14	$z = \sin x - \cos x, x, y \in [-\pi; \pi]$	Анимация. Вращение относительно направления на источник света
15	$z = y^2 \operatorname{sh} x, x, y \in [-2; 2]$	Анимация. Сдвиг вдоль нормали пропорционально времени
16	$x = \sin u, y = v^2, z = u + v,$ $u \in [0; 2\pi], v \in [-1; 1]$	Анимация. Расстояние от вершины до заданной точки меняется по синусоиде
17	$x = u^2 - v^2, y = v^2, z = uv,$ $u, v \in [-2; 2]$	Анимация. Сдвиг вдоль нормали пропорционально времени всех вершин, у которых нормаль составляет с осью $OZ$ острый угол
18	$x = u^3 + v^2, y = v^2, z = u + 2v,$ $u, v \in [-2; 2]$	Анимация. Вращение относительно оси $OZ$ Скорость вращения меняется по синусоиде
19	$x = u^2 - v^2, y = v^3, z = 2u + v,$ $u, v \in [-2; 2]$	Анимация. Изменение интенсивности источника диффузного света по синусоидальному закону
20	$x = u^2 - v^3, y = v, z = u - v,$ $u, v \in [-3; 3]$	Анимация. Изменение цвета источника рассеянного света по синусоидальному закону
21	$x = u^2 + v^2, y = -v^2, z = u,$ $u, v \in [-3; 3]$	Анимация. Изменение яркости поверхности пропорционально косинусу времени
22	$x = u^2 - v, y = v^2 - v, z = 2uv,$ $u, v \in [-3; 3]$	Анимация. Изменение насыщенности поверхности пропорционально косинусу времени

## 7 Визуализация сложных сцен

Разработать формат представления сложной сцены, состоящей из нескольких (текстурированных) объектов, источников света, возможно, камер.

Создать сложную сцену в программе моделирования трехмерной графики (предпочтительно Blender), экспортировать сцену в разработанный формат.

Написать программу, загружающую и визуализирующую построенную сцену.

В присылаемый архив дополнительно необходимо поместить созданную сцену и скрипт для ее экспортирования. В отчет включить описание разработанного формата, визуализацию сцены в программе моделирования и в написанной программе.

*Примечание.* Если созданная сцена обладает нетривиальной геометрией, что влечет за собой необходимость алгоритмической реализации существенных ограничений в перемещении камеры, или созданная программа придает объектам определенную

интерактивность, позволяя менять их местоположение или состояние или даже позволяя уничтожать их, такая работа может быть засчитана в качестве курсовой.

## Курсовая работа

1. Выполнить задание согласно варианту.
  1. Программная реализация буфера глубины (z-буфера).
  2. Программная реализация карт теней (shadow mapping).
  3. Отбрасывание теней методом трассировки лучей.
  4. Трехмерная анимированная модель Солнечной системы.
  5. Реализация прозрачных и зеркальных поверхностей методом обратной трассировки лучей.
  6. Программная реализация методов затенения Гуро, Фонга и Блинна-Фонга.
  7. Программная реализация методов текстурирования тела.
  8. Трехмерная анимированная модель столкновения бильярдных шаров.
  9. Реализация медианного фильтра и постеризации изображения.
  10. Построение трехмерного рельефа по заданной карте глубин и высот.
  11. Построение кусочно-задаваемых B-сплайнов.
  12. Построение кусочно-задаваемых кривых Безье.
  13. Построение NURBS.
  14. Анимация трехмерной модели методом морфинга.
  15. Реализация фильтра выделения границ и произвольного изменения контрастности полученных областей.
  16. Реализация алгоритма Брезенхема для отрезка и окружности, сглаживание.
  17. Реализация булевых операций над многоугольниками.
  18. Текстурирование модели *процедурной картой*.
  19. Программная реализация тумана.

20. Программная реализация модели освещенности с затуханием.
21. Реализация объемного света.
22. Реализация гауссова и контрастоповышающего фильтров.
23. *Свой вариант.*<sup>1</sup>

**2.** Оформить отчет из следующих частей:

- титульный лист;
- введение (1–2 с.);
- теоретическая часть (5–10 с.);
- описание разработанной программы с цитированием важнейших частей исходного кода и графическими результатами работы (5–10 с.);
- заключение (1–2 с.);
- список литературы.

---

<sup>1</sup>Необходимо предварительно согласовать с преподавателем.