Вычислительная геометрия и геометрическое моделирование

*Индивидуальные задания к лабораторным занятиям* (20 ч)

**Общие требования:**

1. Все задания должны быть выполнены в виде законченного работоспособного программного приложения с демонстрацией результатов расчета **в графическом** виде.
2. ОС – любая, язык программирования – произвольный. Для получения максимального доступа к вычислительным ресурсам и гибкого управления памятью настоятельно **рекомендуется использовать** **С++** (в любой редакции). В качестве GUI-платформы можно использовать библиотеку Qt.
3. Интерфейс пользователя должен позволять ввести размер и параметры задачи (число точек, отрезков и т.д.) или ввести эти данные с помощью интерактивного графического построения (например, задать прямоугольное окно с помощью мыши).
4. Для задач по вычислительной геометрии (темы 1, 3) **для демонстрации производительности** размер задачи должен быть большим: не менее 105-106. Для того, чтобы отображение результата не занимало больше времени, чем сам расчет, точки следует отображать просто **пикселями** (а не окружностями или эллипсами). Для динамических изображений перерисовку в окне следует выполнять только для **изменяющейся** части окна. Приложение необходимо запускать в конфигурации **Release**, в потоки std::cout и std::cerr ничего не выводить.
5. В задачах по компьютерной графике (тема 2) построенный 3D объект нужно демонстрировать **в динамике**, например, в виде простого предмета (куба, тетраэдра), вращающегося вокруг своей оси, или в виде статического объекта, вокруг которого движется камера.
6. В задачах по геометрическому моделированию (тема 4) построение кривой или поверхности необходимо производить интерактивно, задавая управляющие точки с помощью мыши.
7. В сумме по всем темам необходимо набрать **30 баллов**. По каждой теме нужно выполнить не менее одного задания. Задания одной темы разделены на группы и выделены цветом, выбранные задания из одной темы должны быть разных цветов.
8. В процессе защиты работы необходимо предъявить код программы и быть готовым ответить на вопросы по коду и алгоритму.

|  |  |  |
| --- | --- | --- |
| **№** | **Баллы** | **Тема 1**: *Геометрический поиск и пересечения* |
|  | 4 | Подсчет точек внутри заданного прямоугольного окна за *O*(log *n*).   * число точек не менее 106, создаются с помощью генератора случайных чисел; * окно задается с помощью мыши после генерации точек; * после построения окна его можно захватить мышью и перемещать; * при каждом перемещении в специальном окне отображается число точек в окне. |
|  | 4 | Перечисление точек внутри заданного прямоугольника: метод сетки.   * число точек не менее 106, создаются с помощью генератора случайных чисел; * окно задается с помощью мыши после генерации точек; * после построения окна его можно захватить мышью и перемещать; * точки, попавшие в окно, выделяются цветом. |
|  | 5 | Перечисление точек внутри заданного прямоугольника: квадратичное дерево (см. задачу 2). |
|  | 5 | Перечисление точек внутри заданного прямоугольника: 2-d-дерево (см. задачу 2). |
|  | 6 | Перечисление точек внутри заданного прямоугольника: адаптивное 2-d-дерево (см. задачу 2). |
|  | 4 | Принадлежность простому многоугольнику.   * многоугольник задается с помощью мыши; * после построения многоугольника он окрашивается в какой-нибудь цвет, если указатель мыши находится внутри, и не окрашивается, если указатель находится снаружи. |
|  | 6 | Принадлежность простому многоугольнику с использованием устойчивых предикатов Шевчука (см. задачу 6). |
|  | 6 | Принадлежность звездному многоугольнику, с поиском внутренней точки (см. задачу 6). |
|  | 6 | Локализация точки на планарном разбиении: метод полос.   * в качестве планарного разбиения использовать прямоугольную сетку размера не менее 1000x1000, повёрнутую на небольшой угол (5-7 градусов); * после построения и отображения сетки при движении курсора ячейка сетки под курсором выделяется цветом. |
|  | 10 | Локализация точки на планарном разбиении: метод детализации триангуляции (см. задачу 9, размер сетки 100x100, ячейки сетки разбиваются на два треугольника). |
|  | 5 | Пересечение выпуклых многоугольников, проверка выпуклости.   * многоугольники задаются с помощью мыши; * проверка выпуклости выполняется с помощью устойчивых геометрических предикатов Шевчука. |
|  | 5 | Пересечение ортогональных отрезков.   * отрезки создаются автоматически с помощью генератора случайных чисел; положение и длина генерируются отдельно, при этом средняя длина должна быть намного меньше области, занимаемой отрезками; * после построения отрезков поиск пересечений запускается отдельной командой (кнопкой); * пары пересекающихся отрезков выделяются цветом. |
|  | 5 | Идентификация пересечения отрезков.   * см. задачу 12. * факт пересечения отрезков проверять с помощью устойчивых предикатов Шевчука. |
|  | 6 | Пересечение отрезков (алгоритм Бентли-Оттмана).   * см. задачу 12. * факт пересечения отрезков проверять с помощью устойчивых предикатов Шевчука. * точки пересечения вычислять с помощью обычной (неточной) арифметики. |
|  | 10 | Пересечение отрезков (алгоритм Бентли-Оттмана) с использованием точной арифметики (библиотека GMP).   * см. задачу 12. * факт пересечения отрезков проверять с помощью устойчивых предикатов Шевчука. * при формировании приоритетной очереди критических точек использовать библиотеку точной арифметики (GMP). |
|  | | |
|  |  | **Тема 2:** *Плоские проекции. Облет или вращение объекта.* |
|  | 5 | Построение изометрии. |
|  | 5 | Построение проекции Cavalier. |
|  | 5 | Построение проекции Cabinet. |
|  | 5 | Построение одноточечной центральной проекции. |
|  | 5 | Построение двухточечной центральной проекции. |
|  | 5 | Построение ортографической проекции по заданному центру проекции и фокусу (нормаль к картинной плоскости направлена на объект, вектор вертикали - вверх). |
|  | 5 | Построение ортографической проекции по заданным центру проекции, фокусу и вектору вертикали (нормаль к картинной плоскости направлена на объект). |
|  | | |
|  |  | **Тема 3:** *Выпуклые оболочки, задачи геометрической близости* |
|  | 4 | Выпуклая оболочка: метод Джарвиса.   * число точек не менее 105, создаются с помощью генератора случайных чисел; * после генерации точек вычисление выпуклой оболочки запускается отдельной командой (кнопкой); * построенная оболочка выделяется цветом. |
|  | 6 | Выпуклая оболочка: метод Джарвиса, устойчивый.   * см. задачу 1; * поиск каждой следующей точки выполнять с помощью устойчивых предикатов Шевчука. |
|  | 4 | Выпуклая оболочка: метод Грехэма (см. задачу 1). |
|  | 6 | Выпуклая оболочка: метод Грехэма, устойчивый.   * см. задачу 1; * при сортировке и обходе Грэхема использовать устойчивые предикаты Шевчука. |
|  | 4 | Выпуклая оболочка: «быстрый» метод (см. задачу 1). |
|  | 6 | Выпуклая оболочка: «быстрый» метод, устойчивый.   * см. задачу 1; * использовать устойчивые предикаты Шевчука. |
|  | 6 | Выпуклая оболочка: метод «разделяй и властвуй».   * см. задачу 1; * использовать устойчивые предикаты Шевчука. |
|  | 4 | Аппроксимация выпуклой оболочки (см. задачу 1). |
|  | 6 | Выпуклая оболочка простого многоугольника.   * см. задачу 1; * использовать устойчивые предикаты Шевчука. |
|  | 6 | Триангуляция Грэхема.   * число точек не менее 500, создаются с помощью генератора случайных чисел для разных типов распределений; * после генерации точек вычисление запускается отдельной командой (кнопкой); * использовать устойчивые предикаты Шевчука. |
|  | 6 | Триангуляция Делоне: flip-алгоритм (см. задачу 11). |
|  | 6 | Триангуляция Делоне: метод «разделяй и властвуй», *O*(*n* log *n*) (см. задачу 11). |
|  | 10 | Диаграмма Вороного: метод «разделяй и властвуй».   * число точек не менее 500, создаются с помощью генератора случайных чисел для разных типов распределений; * после генерации точек вычисление запускается отдельной командой (кнопкой). |
|  | 10 | Диаграмма Вороного: метод Fortune (см. задачу 13). |
|  | 10 | Триангуляция сгущения, алгоритм Ruppert (см. задачу 13). |
|  | | |
|  |  | **Тема 4:** *Геометрическое моделирование* |
|  | 4 | Алгоритм Кастельжо для кривых Безье.   * управляющие точки задаются с помощью мыши, по мере появления новых точек строится кривая; * должна быть возможность переместить управляющую точку в любой момент, вслед за ней меняется форма кривой. |
|  | 4 | Кубические кривые Эрмита (см. задачу 1). |
|  | 5 | Полиномы Эрмита пятой степени (см. задачу 1). |
|  | 5 | Составные кубические кривые в форме Безье (см. задачу 1). |
|  | 5 | Аппроксимационные B-сплайны 3-го порядка (см. задачу 1). |
|  | 5 | Интерполяционные B-сплайны 3-го порядка (см. задачу 1). |
|  | 5 | Кубическая поверхность Безье.   * управляющие точки создаются автоматически, по умолчанию лежащими в одной плоскости и образующими прямоугольную область; * должна быть возможность переместить управляющую точку, вслед за ней меняется форма поверхности. * поверхность отображается в ортогональной проекции (Z = 0) в виде сетки. |
|  | 5 | Кубическая поверхность Эрмита.   * управляющие точки, вектора касательных и нормали создаются автоматически (точки лежат в одной плоскости и образуют прямоугольную область); * должна быть возможность переместить управляющую точку и изменить направление векторов в вершинах, вслед за ней меняется форма поверхности. * поверхность отображается в ортогональной проекции (Z = 0) в виде сетки. |
|  | 6 | Составные кубические поверхности Безье с гладкой сшивкой на границе (см. задачу 7). |
|  | 6 | Составные кубические поверхности Эрмита с гладкой сшивкой на границе (см. задачу 7). |
|  | 6 | Линейные куски Кунса.   * граничные кривые по умолчанию создаются как кубические кривые Безье или Эрмита, образующие замкнутый 4-угольный контур; * изменение формы кривой меняет форму поверхности. * поверхность отображается в ортогональной проекции (Z = 0) в виде сетки. |
|  | 10 | Кубические поверхности Кунса, гладкая сшивка по общей границе (см. задачу 10). |