

Задача Weil – FIT(FF)_GGGG_Familiya_Weil_1

Реализовать алгоритм отсечения Вейлера-Айзертон.

В файле xxxxxx_About.txt четко описать, что реализует программа. Условие:

1. Координаты вершин задаются по целочисленной сетке раstra экрана (клиентской области окна).
2. Многоугольники (М) ввести при помощи мыши либо считать вершины из файла. Учитывается направление обхода, т.е. внутренность М слева (иначе сразу 2). Запись/чтение файла обязательно программируются (иначе сразу 2). В начале работы экран пустой. По кнопке нового документа – также нет модели. В процессе работы можно изменять оба многоугольника.
3. Не забывать спасать текущую модель в файле.
4. Отсекаемый М и отсекающий Э объекты (границы), а также многоугольник-результат (МР) отобразить разными цветами. Для ввода М и Э свои кнопки. Для отображения каждого из 3-х многоугольников использовать толщину линии постоянного значения, возможно, разного. Значит должны быть диалоги задания толщин и цветов.
5. Завести кнопку, по которой выполняется операция отсечения.
6. Результат операции нарисовать третьим цветом, не стирая исходных объектов.
7. По желанию: многоугольники можно отобразить закрашенными (или заштрихованными), т.е. не только границей.
8. ЗИП и директория проекта - FAK_GGGG_Familiya_Weil_1 (или _N)
9. Проект должен быть составлен таким образом, что при выполнении чтения файла мы сразу попадаем в директорию проекта FAK_GGGG_Familiya_Weil_N.
10. В директории FAK_GGGG_Familiya_Weil_Data личный файл(ы)-пример(ы) в заданном формате и с правильными именами. Пусть М (имеет k контуров) – отсекаемый, а Э (имеет t контуров) – отсекающий. Тогда файл имеет следующий вид. На последней строке не должен быть EOF.

```
fx fy    // размер поля в пикселях
k         // число контуров в М
далее идут k групп строк с описанием контуров
t         // число контуров в Э
далее идут t групп строк с описанием контуров
EOF
```

Каждый контур занимает несколько строк, которые имеют вид:

```
n         // число вершин в контуре
xi yi     // i = 1..n на отдельных строках
```

11. x_i y_i задаются в правосторонней системе координат: ось X – слева направо, ось Y – **снизу вверх**. (0,0) – левый нижний угол.
12. f_x f_y – поле рисования в пикселях. После чтения файла устанавливать такой размер клиентской области окна. Должна быть кнопка, позволяющая изменить размеры поля. В том случае, когда в поле заданного размера не входит уже имеющийся рисунок, давать сообщение, а операцию игнорировать.
13. Задача сдается в 1 прием по 14 марта 2011 года.
14. Просьба ознакомиться с алгоритмом и задавать вопросы в форуме, также готовить вопросы к пятницам.
15. Специфические случаи взаиморасположения многоугольников и контуров (касание сторон, совпадение вершин и т.п.) можно не рассматривать, объяснено на лекции.

Многоугольники. Рассмотрим определение многоугольника, на которое мы будем опираться. **М** – это связная область (возможно с дырками), т.е. один внешний контур и, возможно, несколько внутренних. Если каждой стороне приписать направление и все стороны упорядочить, т.е. создать из них ориентированный путь (контур), то *положительной* считается ориентация, когда обход границы М (контуров) делается **против часовой стрелки** – область остается слева.

Внимание.

Упрощение задачи: М имеет *ровно* один внутренний контур или *ни одного*. Э не имеет дыр (внутренних контуров), т.е. односвязный. Результат МР: в общем случае – это несколько многоугольников. **НЕ делайте вариант, когда Э многосвязный.**

Несколько замечаний по реализации задачи

1. При обходе границы область находится слева. Это не всегда соблюдается исполнителями. Левый обход – грубая ош.
2. Для замыкания многоугольника необходимо чтобы курсор мыши попал в некоторую окрестность 1-й вершины, тогда М замкнется. Некоторые предпочитают давить правую клавишу мыши. В этом случае делается замыкание последней из построенных вершин с первой. В описании указать как у вас, кратко: «замыкание по правой кнопке мыши», «замыкание контура по попаданию курсора в окрестность 7 пикселей».
3. Следить, чтобы пользователь не добавлял стороны, пересекающие предыдущие построенные стороны.

Удобно иметь кнопки меню:

Отсекаемый – затем строится

Дырка – затем строится

Отсекающий – затем строится

Выполнение отсечения

Внимание:

Обязательно проверить случаи, когда М содержится внутри Э, когда Э содержится внутри М.

Дополнительно для желающих:

Смена ориентации **односвязных** М и Э, т.е. реализуются ТМО.

Не давать строить контура с «правым» обходом области

Примеры:

- Строт – допускает самопересечения контуров. И, видать, пользуется к-л внутренней процедурой. А реализация алгоритма должна быть своя. Так не делать.
- Козлов – так работает алгоритм Сазерленда.
- Рябков – пример Вейлера с теоретико-множественными операциями, т.е. смена ориентации обхода.