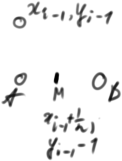
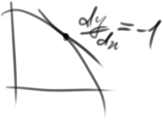
19.03.14-----------------------------------------------------------------------



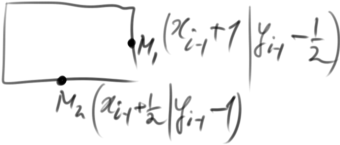


*.*

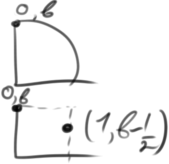
. A4y2=b4x2 (1), отсюда

. Отсюда имеем , т.е.

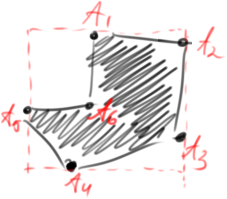
Необходимо также сделать вторую коррекцию на границе интервалов. Сделаем её однократно, в отличие от предыдущей, которая выполнялась двукратно.



Наконец надо вычислить функцию в начальной точке.



**Растровая развёртка сплошных областей (заливка).**

Генерация сплошных областей на базе простых описаний вершин или рёбер многоугольника.

1 границы можно не отрисовывать; принадлежат внутренней области

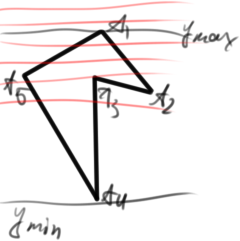
2 Алгоритм, справляющийся с произвольным многоугольником (выпуклые/невыпуклые; внутри могут содержаться границы

Фактически, задача сводится к определению принадлежности каждого пикселя внутренней области, ограниченной прямоугольник. Это определяется путём «запуска вектора». Если он пересекает прямоугольник дважды – это закрашенная область, единожды – вершина.

Условия: xmin <= x <= xmax && ymin <= y <= ymax.

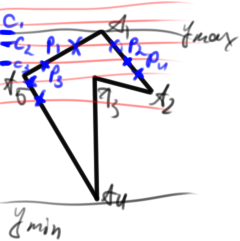
Если область, которую нужно закрасить, значительно меньше прямоугольника граничных координат, то КПД простейшего алгоритма чрезвычайно мал.

Все алгоритмы заполнения можно разделить на две группы – растровые и затравочные.

В растровом алгоритме для каждой строки каждый раз решается задача определения, лежит пиксель внутри многоугольника или нет. Рассматриваются сканирующие строки, пересекающие многоугольник.

Для пикселей каждой сканирующей строки определяется их принадлежность многоугольнику.

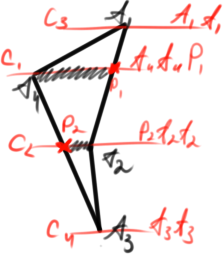
Уменьшить трудоёмкость задачи можно если принять во внимание факт того, что пиксели, расположенные друг рядом с другом, сохраняют неизменным «свойство принадлежности».



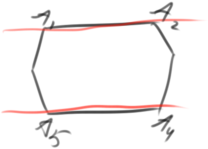
С1: [A1..A1]

C2: [P1..P2]

C4: [P3..A3] [A3..P4]

Используя подход, связанный с учетом когерентности пикселей (одинаковые свойства рядом расположенных пикселей), проблема возникает только для строк, проходящих ровно через вершины многоугольника. Проблема: для одних строк количество пересечений надо принимать ==2, а для других ==1

Если в примере выше взять ребра А1А5 и А1А2, то у1>у5 && у1>у2. Аналогично, y4<y5, y4<y3. А вот для пятой вершины, y5<y1 и y5>y4 – точка А5 не является точкой экстремума; в то время как А1 и А

Вторая проблема – с горизонтальными рёбрами. Ребро лежит на сканирующей строке – огромное число «пересечений. Решение – выкинуть ребро. Тогда сканирующая строка всёравно будет пересекаться с А1.. но уже как с концами других рёбер – это не значит, что они не будут закрашены.