平面上の点集合を反時計回りに整列

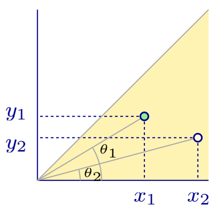
Stackoverflow:
Fastest way to sort vectors by angle without actually computing that angle

↑ 超絶すばらしくって参考になるサイト

こっちも重要 ↓

[参考] ファウラーの擬角
スティーブさんの C コード

タンジェントとコタンジェント

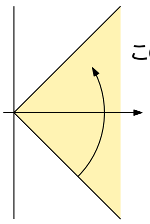


事象
$\theta_1 < \theta_2$

ついでに
$\tan\theta_1 < \tan\theta_2$
$\frac{y_1}{x_1} < \frac{y_2}{x_2}$

黄色の領域に配置される点の tan の値域は $[0, 1]$

ちょっと視野を広げて

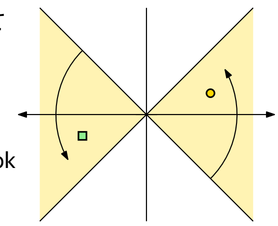


この領域内の点の tan は $\frac{y}{x} \in [-1, 1]$ で計算できて

この値で点間の大小を測ることができる

ここの基準線 $y = -x$ が最小値と設定する

もうちょっと視野を広げて



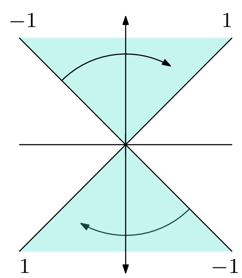
こっちの領域内の2点間の比較も $\frac{y}{x}$ で ok
大小の順は矢印の雰囲気

○ と □ の大小関係はもっとメタに解決できる!!

例えば対角線 $y = x$ と $y = -x$ を基準に
$y_1 > x_1$ かつ $y_1 < -x_1$ なら右の三角に属すし
$y_2 \leq x_2$ かつ $y_2 \geq -x_2$ なら左の三角に属すので
$\theta_1 < \theta_2$ が言える　(ふんいき理解! ok???)

さらに値域が発散しないよう $[-1, 1]$ に限定するために cotan $= \frac{x}{y}$ を用いれば

下図のように $\times -1$ と $-x$ にすれば領域内の大小比較が可能となる

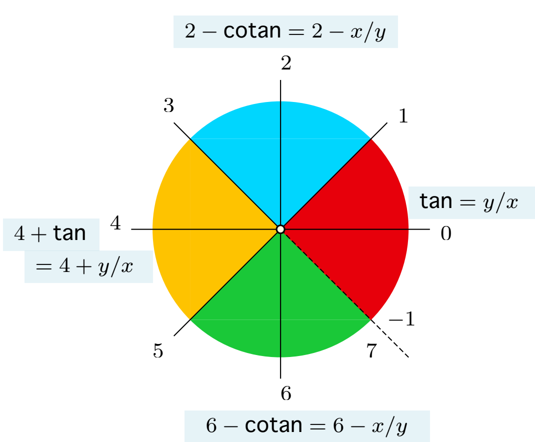


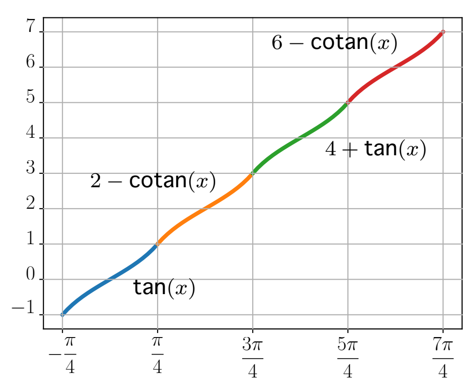
原点は整列対象に含まれないこと!

これ、大前提です!!

チンバリさんのやり方

下図のような感じでラジアン $[0, 360]$ を $[-1, 7]$ へ移す手続きをわずかな回数の四則演算で実現する





こんな感じで定義域を4分割しつつ tan と cotan を
使い分けつつ組合せて関数を定義すると

「連続」で「単調増加」な性質を有することが確認できる

$x = 0$ や $y = 0$ のときは「0 割り」が入るので少し注意

※ メタ分類で排除できる懸念ではある!

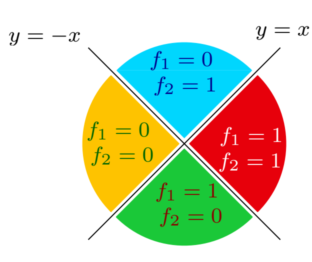
※ 繋ぎ目不連続に見えるけど y/x だったり $-x/y$ なので微分可能ではありますよね

平面のメタ四分割

チンバリさんの平面分割の4つの部分領域にコードを付与

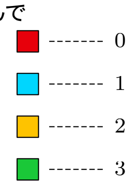
$$f_1(x, y) = \begin{cases} 1 & \text{if } y < x \\ 0 & \text{otherwise} \end{cases}$$

$$f_2(x, y) = \begin{cases} 1 & \text{if } y > -x \\ 0 & \text{otherwise} \end{cases}$$



各エリアを $f_1 + 2f_2$ およびなんら間お合成関数 で符号化したい

んで



まで翻訳できればメタな比較で
角度の大小が判定できる

もし比較対象の2点が
同一領域に配置された場合は

コードが偶数の場合は $\frac{y_1}{x_1} \overset{?}{>} \frac{y_2}{x_2}$

奇数なら $-\frac{x_1}{y_1} \overset{?}{>} -\frac{x_2}{y_2}$ で大小判定可!

jjrv さんの解法コード

```
qa = 2 * (x1 > y1) + (x1 > -y1)
qb = 2 * (x2 > y2) + (x2 > -y2)
if qa != qb:
    return ((0x6c >> qa * 2 & 6)
                - (0x6c >> qa * 2 & 6))
```

0x6c は 108 で二進コードは 0b1101100

ちょっと発想が素晴らしいですよね。