

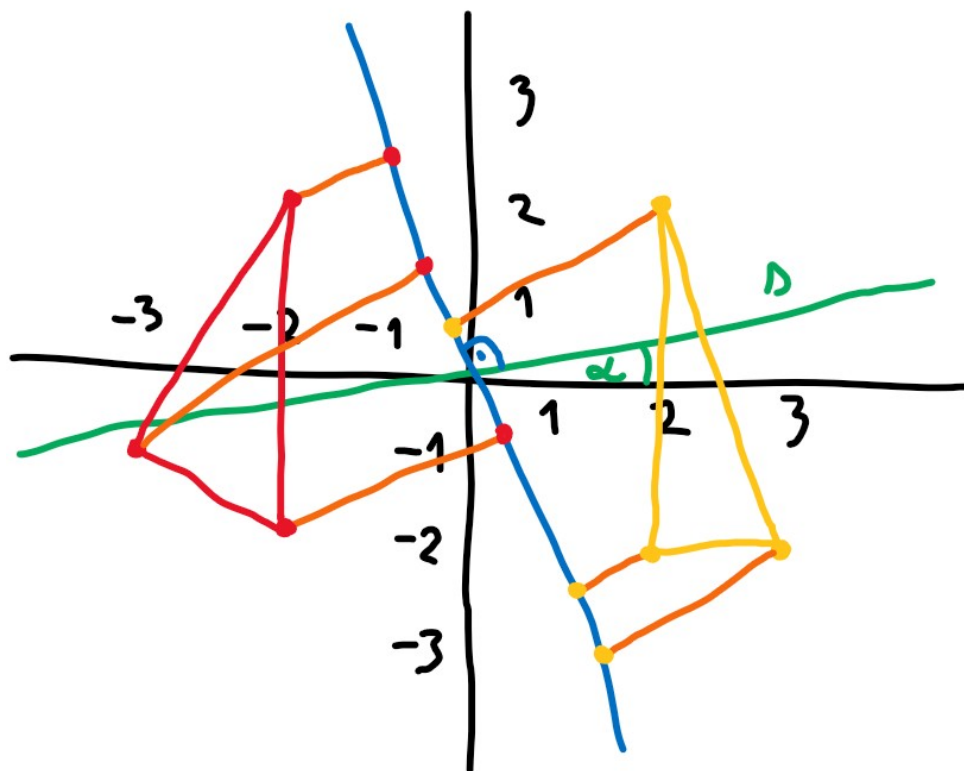
Ryby

Transformácia mnohouholníkov na kolmicu smeru

Úhol smeru (α) môžeme vypočítať ako arcus tangens koordinátov smeru s_x a s_y . Na tento smer spravíme kolmicu (modrá), ktorá prechádza bodom o súradniciach $[0, 0]$. Na túto kolmicu (modrá) budeme premietiť každý bod každého mnohouholníka.

Po premietnutí všetkých bodov mnohouholníka nájdeme dva najvzdialenejšie body a zapíšeme ich do hash mapy, kde k nim priradíme príslušný mnohouholník.

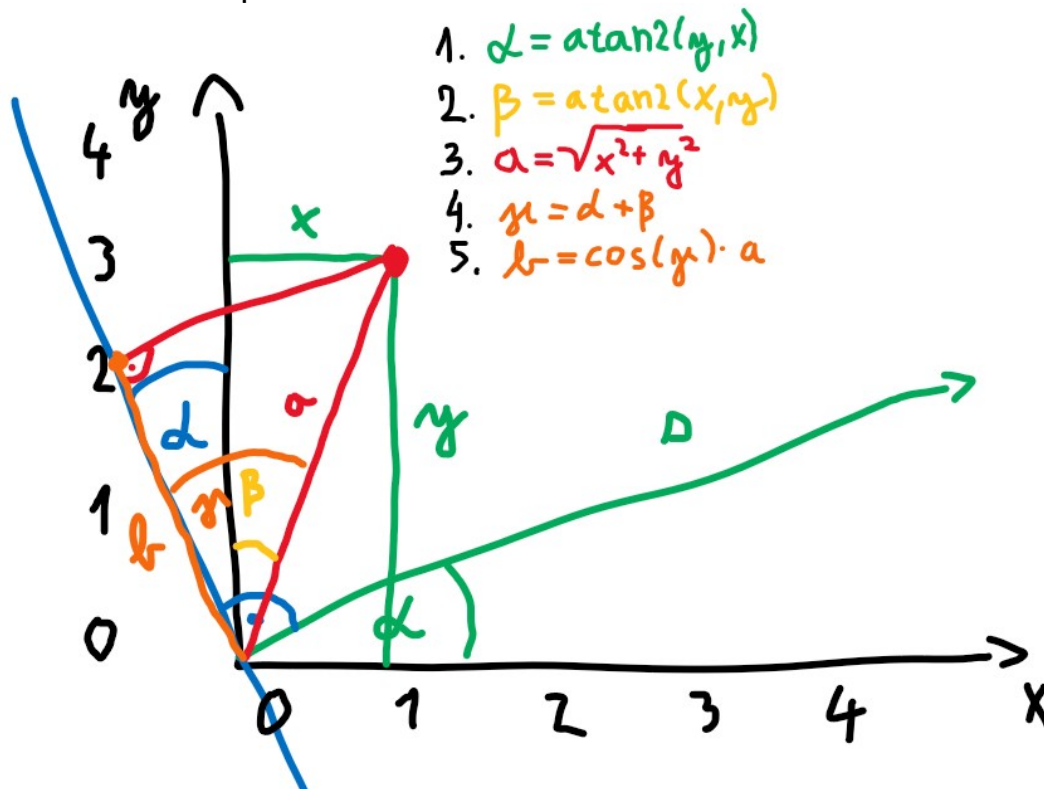
Náčrt názorného premietnutia mnohouholníkov je na Obr. 1. Na obrázku máme znázornené 2 trojuholníky (červený a žltý), ktoré sa premietnu na kolmicu (modrá) ako dve úsečky. Úsečky sa získajú ako minimálna a maximálna hodnota bodov na priamke príslušného mnohouholníka.



Obr. 1.

Príklad transformácie jedného bodu na kolmicu smeru (postup, Obr. 2)

1. výpočet uhlu α (smeru pohybu lode), úhol α vypočítam pomocou hyperbolickej funkcie tangens (cotangens), kvôli tomu, aby som mal správny smer pre všetky 4 kvadranty, pričom kvadranty 1, 3 a 2, 4 sú identické z hľadiska smeru
2. výpočet uhlu β (žltý), úhol vypočítame pomocou arcus tangenc polohy bodov x, y , v tomto prípade je nutné použiť funkciu atan2 , aby bol správne definovaný kvadrant polohy bodu
3. výpočet uhlopriečky a (červená) obdĺžnika x, y
4. výpočet uhlu $\gamma = \alpha + \beta$
5. výpočet pozície b (oranžová) transformovaného bodu na kolmici (modrá) na smer pomocou funkcie \cos uhla γ



Obr. 2.

Nájdenie prieniku najväčšieho počtu premietnutých mnohoúhelníkov

Najprv spravím z hash mapy zoradené pole podľa pozície bodov. Pole následne prejdem a budem si pamätať prienik najväčšieho počtu premietnutých mnohoúhelníkov. V aktuálnom prieniku sú vždy všetky premietnuté mnohoúhelníky, ktorých prvý bod som už prešiel, ale neprešiel som ich druhý bod. Aby nedošlo k chybe, vždy keď v jednom bode sú okraje viacerých premietnutých mnohoúhelníkov, najprv pripočítam všetky body a následne odpočítam body mnohoúhelníkov, ktoré sa v tomto bode končia (aby neboli započítané 2-krát). Aktualizujem maximálny prienik (ak je to potrebné) a potom odpočítam od aktuálneho prieniku počet mnohoúhelníkov, ktoré sa v tomto bode končia.

Časová zložitosť algoritmu

Asymptotická zložitosť algoritmu je $O(n \log n + m)$, kde n je počet mnohoúhelníkov a m je súčet bodov všetkých mnohoúhelníkov. Premietnutie všetkých bodov mnohoúhelníkov sa udeje vždy s

časovou zložitost'ou $O(2m)$ kvôli tomu, že na premietnutie potrebujeme nájsť 2 najvzdialenejšie body z každého mnohoúhelníka. Nájdenie prieniku najviac premietnutých mnohoúhelníkov je $O(2n(\log(2n) + 1))$, lebo každý premietnutý mnohoúhelník má 2 krajné body a všetky body treba zoradiť a následne nájsť maximálny prienik. To je spolu $O(2n(\log(2n) + 1) + 2m)$, čo je po odstránení konštánt $O(n \log n + m)$.

Prestnostné chyby

Hoci polohy bodov mnohoúhelníkov sú zadávané v celých číslach, výpočet uhlov, uhlopriečky a premietnutého bodu sú čísla desatinné. Pri výpočte konkrétneho desatinného čísla nastáva priblíženie k jeho skutočnej hodnote. Chyba sa znásobuje pri používaní aritmetických operácií rádovo. Konkrétne, ak chyba nastane na 15 desatinnom mieste po niekoľkých aritmetických operáciách s podobnými číslami, sa chyba môže posunúť aj na 13 desatinné miesto.

Napríklad pri smere s uhlom 45° premietne bod so súradnicami $[1, 1]$ na $1.7 * 10^{-17}$, aj keď presné premietnutie by malo byť na 0. Chybu preto korigujem zaokrúhlením na 6 desatinných miest.

Zdôvodnenie správnosti algoritmu

Pri transformovaní bodov, sa každý bod premietne na bod na kolmici smeru, cez ktorý by musela priamka prechádzať ak by pretla tento bod mnohoúhelníka. Dve najvzdialenejšie transformované body daného mnohoúhelníka teda tvoria úsečku, pre ktorú platí, že ak by priamka smeru prechádzala cez jej akýkoľvek bod, pretla by daný mnohoúhelník. Hľadám tak úsek na priamke, na ktorom leží najväčší počet úsečiek, teda úsek, ktorý by pretla priamka čo by pretla najväčší počet mnohoúhelníkov.

Záver

Program dokáže vypočítať smer plavby vo všetkých štyroch kvadrantoch a taktiež polohy bodov mnohoúhelníkov môžu byť v ľubovoľnom kvadrante (aj záporné súradnice polohy bodov mnohoúhelníka).