Univerzita Karlova

Přírodovědecká fakulta

Úvod do programování

Výpočet souřadnic průsečíku dvojice úseček (2D)

Zkouška

Adam Bartůšek 3. ročník BGEKA 10.02.2022

1. ZADÁNÍ

Máme dvě úsečky zadané koncovými body p1, p2, p3 a p4. Úkolem programu je analyzovat, zdali se úsečky protínají, a pokud ano, jaké jsou souřadnice jejich průsečíku.

2. POPIS A ROZBOR PROBLÉMU + VZORCE

Problém této úlohy spočívá v poměrně velkém počtu možností vzájemných poloh dvou úseček. Záleží na více proměnných, jsou li úsečky rovnoběžné či různoběžné, pokud jsou rovnoběžné tak leží-li na jedné přímce či nikoli, kolik mají společných průsečíků, je-li alespoň jedna svislá a také která z nich se v programu zadá jako první.

Xové a ypsilonové hodnoty vektorů obou přímek se vypočtou:

vect1x = x2-x1

vect1y = y2-y1

vect2x = x4-x3

vect2y = y4-y3

Základními vzorci jsou vzorce pro výpočet koeficientů a hodnot v obecné rovnici přímky dané dvěma body:

A1 = y2-y1

B1 = x1-x2

C1 = A1*x1+B1*y1

A2 = y4-y3

B2 = x3-x4

C2 = A2*x3+B2*y3

kde xi a yi jsou xové, respektive ypsilonové souřadnice bodu pi a Ai, Bi a Ci jsou koeficienty, resp. konstanty v obecných rovnicích i-té přímky.

Následně se vypočítá determinant matice vzniklé ze soustavy lineárních rovnic podle vzorce det = A1*B2 - A2*B1.

Dalším vzorcem je vzorec na testování, zda leží rovnoběžné přímky (úsečky) v jedné linii:

y1 + z*(y2-y1) = y3, kde z = (x3-x1)/(x2-x1)

Posledním vzorcem je vzorec na výpočet průsečíku přímek:

rx = (B2*C1 - B1*C2)/det

ry = (A1*C2 - A2*C1)/det

3. POPISY ALGORITMŮ FORMÁLNÍM JAZYKEM

Základem programu je tvorba dvou lineárních rovnic o dvou neznámých ze čtyř zadaných bodů dvou úseček. Nejprve je potřeba z rozdílů dvou bodů odečíst vektor. Následně je vytvořen normálový vektor prohozením souřadnic x a y vektoru a vynásobením jedné z nich číslem -1. Tudíž A1 = y2-y1 a B1 = x1-x2, dosazením bodu p1(x1,y1) do rovnice C1 = A1*x1+B1*y1 získáme i hodnotu C1. Stejný postup je proveden u druhé úsečky.

Dále je z dvou vzniklých rovnic vytvořena soustava dvou rovnic o dvou neznámých, která je převedena do matice a je vypočítán její determinant det = A1*B2 - A2*B1. Pokud je matice regulární, tzn. determinant je nenulový, pak se jedná o lineárně nezávislé vektory (úsečky jsou různoběžné). Pokud je matice singulární, má nulový determinant, jedná se o lineárně závislé vektory a úsečky jsou rovnoběžné.

Pokud je matice regulární, pak se pomocí Cramerova pravidla určí průsečík přímek daných body a vektorem:

px,py = (B2*C1 - B1*C2)/det, (A1*C2 - A2*C1)/det

Výrazy v závorce jsou determinanty matic vzniklých přidáním sloupce C1 C2 místo i-tého sloupce původní matice. Sloupec C1 C2, pokud je matice regulární a systém má tedy právě jedno řešení, je lineární kombinací předešlých sloupců a nezvýší se tak hodnost matice.

Po zjištění vzájemné polohy přímek je nutné testovat, zda mají úsečky průsečík a pokud ano, nachází-li se na obou úsečkách.

4. PROBLEMATICKÉ SITUACE A JEJICH ROZBOR + JEJICH OŠETŘENÍ V KÓDU

Jako problematický se jeví vysoký počet různých možností vzájemné polohy úseček. Také to, v jakém pořadí jsou úsečky zadány a které body jsou zadány jako první (resp. 3.), značně komplikuje a prodlužuje celý kód. To je v kódu ošetřeno mnoha výjimkami a mimořádnostmi.

Dalším problémem, který byl shledán téměř neřešitelným, jsou chybné výpočty počítače, kdy po delším výpočtu, který má vyjít např. 50.0, vyjde 50.000000000000000001, což může být problém ve chvíli, kdy je potřeba porovnat vypočtenou souřadnici se souřadnicí stávající. To bylo vyřešeno použitím funkce *round*, která obě souřadnice zaokrouhlí na stejný počet desetinných míst (9). Je ale šance, i když téměř nulová, že zaokrouhlení povede ke špatnému výsledku.

5. VSTUPNÍ DATA, FORMÁT VSTUPNÍCH DAT, POPIS

Vstupními daty jsou čtyři dvojice souřadnic, tedy čtyři body. Souřadnice jsou ve formátu *int* nebo *float*.

6. VÝSTUPNÍ DATA, FORMÁT VÝSTUPNÍCH DAT, POPIS

Výstupními daty je v případě protnutí obou úseček dvojice souřadnic průsečíku, v případě žádného průsečíku hodnota *None* a v případě, že je nekonečně mnoho průsečíků hodnota *inf*.

7. DOKUMENTACE

Program se sestává z jedné funkce *segments_intersection*, která má čtyři argumenty p1, p2, p3 a p4. Pokud má jedna z úseček totožné koncové body, funkce vrátí hodnotu *None.* Následuje získávání jednotlivých souřadnic z bodů, tvoření dvojic pouze xových a pouze ypsilonových souřadnic, definice vektorů a definice výše zmíněných rovnic a výpočet determinantu.

Poté se program dělí na hlavní dvě větve.

První, komplikovanější, hledá průsečík v případě, že determinant = 0, tedy když jsou úsečky rovnoběžné. Poté se větev dělí na případy, kdy jsou úsečky nesvislé, a kdy jsou svislé. Dále se dělí podle toho, zda leží úsečky v jedné linii či ne a konečně i podle počtu průsečíků.

První je případ, kdy jsou úsečky nesvislé, tím pádem x1 != x2. Pokud platí, že z = (x3-x1)/(x2-x1) a zároveň y1 + z*(y2-y1) = y3 (viz výše), jsou úsečky nesvislé a leží obě na jedné přímce. Poté už se jen testuje, zda se úsečky vůbec neprotínají (větší z xových souřadnic krajních bodů první úsečky je menší než menší z xových souřadnic krajních bodů druhé úsečky nebo naopak – podle toho, která z úseček je zadána jako první), zda mají právě jeden průsečík (větší z xových souřadnic krajních bodů první úsečky je větší než větší z xových souřadnic krajních bodů druhé úsečky a zároveň menší z xových souřadnic krajních bodů první úsečky se rovná větší z xových souřadnic krajních bodů druhé úsečky nebo naopak - podle toho, která z úseček je zadána jako první – a zároveň jsou některé dva body totožné) nebo zda mají nekonečně mnoho průsečíků (to nastane ve všech zbylých případech). Pokud jsou úsečky nesvislé a neleží na jedné přímce (jsou kolineární), pak neexistuje žádný průsečík.

V druhém případě jsou úsečky svislé. Tento případ je speciální v tom, že nelze porovnávat xové souřadnice, jako tomu bylo u případu předchozího. Obě úsečky jsou svislé, pokud x1 == x2. Na jedné přímce leží, pokud x1 == x2 == x3 (== x4). Dále se zde opět rozlišují případy žádného, právě jednoho a nekonečně mnoha průsečíků. Postup je analogický jako u nesvislých rovnoběžných úseček na jedné přímce, akorát se pracuje s ypsilonovými souřadnicemi. Opět, pokud úsečky neleží na jedné přímce, nemohou mít žádný průsečík.

V druhé větvi, tedy pokud je determinant nenulový, se podle Cramerova pravidla (viz výše) vypočítají souřadnice průsečíku přímek. Poté se větev dělí na případy, kdy jsou obě úsečky nesvislé, první je svislá, a druhá je svislá, přičemž každý z těchto případů se ještě dělí na právě jeden a žádný průsečík obou úseček.

Pokud x1 != x2 a x3 != x4, není ani jedna z úseček svislá. Poté už stačí jen zjistit, zda leží souřadnice průsečíku px a py na úsečkách, tedy zda jsou větší nebo rovny menším z příslušných souřadnic krajních bodů úseček a menší nebo rovny větším z příslušných souřadnic krajních bodů úseček. Pokud tomu tak není, úsečky nemají žádný průsečík. Stejný postup se použije i v případech, kdy je jedna úsečka svislá, avšak u svislé úsečky se místo xových hodnot musí použít hodnot ypsilonových.

V každém z předešlých případů jsou vráceny buď souřadnice průsečíku, nebo *None/inf* (podle výsledku) a je vypsána hláška, jakou polohu úsečky mají. Nakonec je funkce zavolána.

8. ZÁVĚR, MOŽNÉ ČI NEŘEŠENÉ PROBLÉMY, NÁMĚTY NA VYLEPŠENÍ

Tento program analyzuje vzájemnou polohu dvou úseček, a pokud mají právě jeden průsečík, pak jej nalezne. Problémové je velké množství případů, které je nutno ošetřit. Dalším problémem je nepřesnost výpočtů, kdy při násobení a dělení občas z neznámého důvodu vyjde špatný výsledek (lišící se od správného třeba na 11. desetinném místě). To má za následek chybné porovnávání souřadnic v případech s jedním průsečíkem. To sice bylo vyřešeno zaokrouhlováním, ale bylo by potřeba to asi ještě zlepšit. Také by nejspíš šlo celý problém nějak zjednodušit.

9. SEZNAM LITERATURY

KRUPKOVÁ, O., 2008: Lineární Algebra 1. Univerzita Palackého, Přírodovědecká fakulta, katedra informatiky. Olomouc, 2008. http://phoenix.inf.upol.cz/esf/ucebni/Algebra.pdf