

# Programovanie (1) v C/C++ 2025/26

## Cvičenia 4, miniprojekt

### Mnohouholník

Tento **miniprojekt** je väčšia úloha, na ktorú máte viac času ako na bežné príklady z cvičení, ale máte za ňu aj viac bodov. Začnite na nej pracovať v predstihu, nenechávajte si ju na poslednú chvíľu.

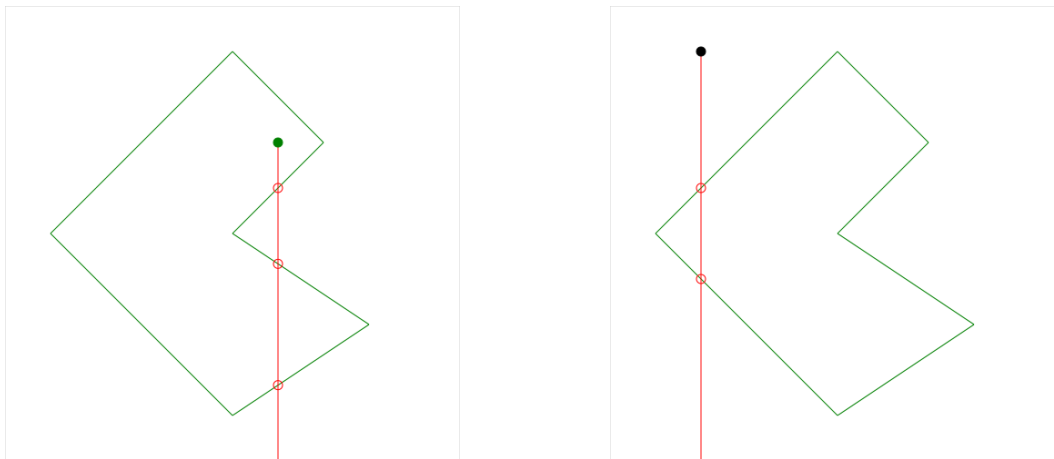
Na miniprojekte pracujte **samostatne**, neukazujte svoj program spolužiakom a nepoužívajte nástroje umelej inteligencie. V prípade otázok kontaktujte vyučujúcich predmetu.

Študenti, ktorí majú z testu pre pokročilých uznané cvičenia 4, budú mať uznaný aj tento miniprojekt a nemusia ho odovzdávať.

Cieľom tejto úlohy je precvičiť si prácu s poliami, structami, ako aj s ďalšími základnými stavebnými prvkami jazyka C++. Budeme riešiť zaujímavý a prakticky dôležitý príklad z geometrie, ale nemusíte sa báť, potrebná matematika je v zadaní vysvetlená. Zadanie je však pomerne dlhé, trénujeme teda aj prácu s odborným textom. Zadanie si poriadnej prečítajte a dodržujte všetky požadované detaily.

V tejto úlohe budeme uvažovať mnohouholník  $P$  zadaný súradnicami svojich vrcholov  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ . Navyše máme danú množinu  $A$ , ktorá obsahuje  $m$  bodov, každý tiež so známymi súradnicami. Cieľom je pre každý bod z  $A$  spočítať, či patrí do vnútra mnohouholníka  $P$  alebo či je mimo jeho hraníc. Napríklad si môžete predstaviť, že mnohouholník reprezentuje hranice štátu alebo iného územia a body z množiny  $A$  predstavujú miesta zoskoku parašutistov a chceme zistiť, koľkým parašutistom sa podarilo pristáť v cieľovom území.

Túto úlohu budeme riešiť nasledovným spôsobom: máme bod  $(x, y)$ , o ktorom chceme vedieť, či je vo vnútri mnohouholníka. Spustíme z tohto bodu polpriamku smerom zvislo nadol a spočítame, s koľkými stranami mnohouholníka sa táto polpriamka pretína. Ak je toto číslo nepárne, bod je vo vnútri, ak je číslo párne, bod je vonku z mnohouholníka. Na obrázku nižšie vidíte naľavo príklad bodu, ktorý je vo vnútri a ktorého polpriamka má tri priesečníky, vpravo bod, ktorý je vonku a jeho polpriamka má dva priesečníky. V oboch prípadoch sú priesečníky zvýraznené červeným krúžkom. Pozor, bod  $(0,0)$  je v ľavom hornom rohu obrázku, os  $y$  ide nadol.



Obr. 1: Dve ukážky polpriamky a jej priesečníkov.

Informácie o výsledkoch programu budeme vypisovať na konzolu, ale aj vykresľovať do obrázku pomocou knižnice SVGdraw.

**Vstup:** Váš program by mal vstup načítavať z konzoly. Na prvom riadku budú celé čísla  $n$ ,  $m$  a  $d$ . Číslo  $n$  je počet vrcholov mnohouholníka,  $m$  je počet bodov v množine  $A$  a  $d$  je určuje veľkosť (šírku aj výšku) výsledného

obrázku.

Na každom z nasledujúcich  $n$  riadkov sú súradnice jedného vrcholu mnohoúhelníka  $P$  a potom nasleduje  $m$  riadkov so súradnicami jednotlivých bodov v množine  $A$ . Všetky súradnice budú reálne čísla.

Môžete predpokladať, že vstup spĺňa nasledovné predpoklady:

- $1 \leq n, m \leq 100$  a  $d > 0$
- Všetky súradnice sú vo vnútri obrázku, t.j. medzi 0 a  $d$ .
- Žiadne dva susedné vrcholy mnohoúhelníka nemajú rovnakú  $x$ -ovú súradnicu (t.j. žiadna strana mnohoúhelníka nie je zvislá) a žiaden bod množiny  $A$  nemá rovnakú  $x$ -ovú súradnicu ako vrchol mnohoúhelníka. Taktiež žiaden bod množiny  $A$  nie je presne na hranici mnohoúhelníka.

Posledná podmienka nám uľahčí programovanie, pretože nemusíme uvažovať okrajové prípady, kedy polpriamka z bodu  $(x, y)$  prechádza presne cez vrchol mnohoúhelníka a treba sa správne rozhodnúť, či to započítavať ako priesečník.

**Výstup:** Váš program by mal na konzolu vypísať jeden riadok pre každý bod z množiny  $A$ , v takom poradí, ako boli na vstupe. Výstup bude obsahovať najskôr slovo **ano** alebo **nie**, podľa toho, či príslušný bod  $(x, y) \in A$  leží v mnohoúhelníku. Ďalej bude obsahovať počet určujúci, koľko strán mnohoúhelníka má priesečník s polpriamkou vedúcou nadol z príslušného bodu (podľa parity tohto počtu sme určovali, či je bod v mnohoúhelníku). Údaje vypisujte vo formáte ako v príklade nižšie.

**Vykresľovanie:** Obrázok uložte do súboru `body.svg` (nepridávajte cestu, t.j. meno priečinku). Súradnice bodov zo vstupu použite priamo ako súradnice v obrázku. Mnohouhelník vykreslite pomocou čiar zelenej farby. Body z množiny  $A$  vykreslite ako krúžky s polomerom 5 a stredom na príslušných súradniciach. Ak je bod vo vnútri mnohoúhelníka, vykreslite ho zelenou farbou, ak je vonku, vykreslite ho čiernou farbou.

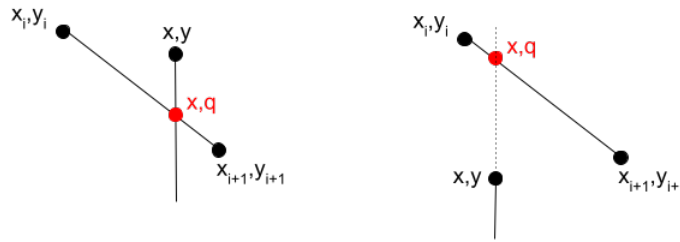
Pre prvý bod na vstupe navyše nakreslite zvislú polpriamku z tohto bodu až po spodný okraj obrázka a každý priesečník tejto polpriamky so stranami mnohoúhelníka zvýraznite krúžkom červenej farby, ktorého polomer je 5 a stred je v danom priesečníku. Pre ostatné body z množiny  $A$  už polpriamky nekreslite, aby nebol obrázok neprehľadný.

**Ako nájsť priesečník:** Uvažujme polpriamku nadol z bodu  $(x, y)$  a jednu stranu mnohoúhelníka spájajúcu vrcholy  $(x_i, y_i)$  a  $(x_{i+1}, y_{i+1})$ . Aby polpriamka pretínala túto úsečku, musí byť predovšetkým súradnica  $x$  niekde medzi súradnicami  $x_i$  a  $x_{i+1}$  (pozor, nevieme, ktoré z  $x_i$  a  $x_{i+1}$  je menšie). Ak je táto spomienka splnená, môžeme spočítať, na akej  $y$ -ovej súradnici pretína zvislá priamka idúca cez  $(x, y)$  túto úsečku. Nazvime túto  $y$ -ovú súradnicu priesečníku  $q$ . Hodnotu  $q$  potom porovnáme s  $y$ , aby sme zistili, či je nad alebo pod bodom  $(x, y)$  a teda či je na našej polpriamke. Hodnotu  $q$  spočítame týmto vzorcom (dá sa odvodiť z podobnosti trojuholníkov):

$$q = y_i + (x - x_i)(y_{i+1} - y_i)/(x_{i+1} - x_i)$$

**Ďalšie požiadavky:**

- Na ukladanie súradníc použite `struct bod` z prednášky 5.
- Naprogramujte a použite funkciu `priesečník`, ktorá dostane súradnice jedného bodu ako aj súradnice dvoch susedných vrcholov mnohoúhelníka a určí, či má polpriamka nadol zo zadaného bodu priesečník s úsečkou spájajúcou zadané vrcholy mnohoúhelníka. Pokiaľ priesečník existuje, funkcia tiež uloží jeho súradnice do ďalšej premennej typu `bod`, ktorú dostane ako parameter odovzdaný referenciou.
- Okrem správnosti budeme pri bodovaní prihliadať aj na pekný programátorský štýl. Váš program by mal byť prehľadný, so správne odsadenými riadkami a rozumnými menami premenných a funkcií.



Obr. 2: Vľavo: situácia keď  $(x, q)$  je pod  $(x, y)$  a teda polpriamka pretína úsečku. Vpravo: polpriamka nepretína úsečku.

- Testovač bude kontrolovať iba správnosť výstupu na konzolu, nezabudnite si ručne skontrolovať vygenerovaný obrázok.

### Prehľad potrebných príkazov z knižnice SVGdraw

- `#include "SVGdraw.h"`
- `SVGdraw drawing(velkost, velkost, "body.svg")` inicializuje obrázok zadaných rozmerov
- `drawing.drawLine(x1,y1,x2,y2)` vykreslí čiaru medzi dvoma zadanými bodmi
- `drawing.drawEllipse(stredX, stredY, polomer, polomer)` nakreslí kruh s daným stredom a polomerom
- `drawing.setLineColor("green")` nastaví farbu čiary na zelenú. Okrem toho používajte aj farby "black" a "red"
- `drawing.setFillColor("green")` nastaví farbu na vyfarbovanie kruhov na zelenú, podobne "black"
- `drawing.setNoFill()` zruší vyfarbovanie, ďalšie kruhy bude kresliť prázdne, kým sa opäť nespustí `drawing.setFillColor`
- `drawing.finish()` správne ukončí vypisovanie obrázku

Kružky pre body kreslite aj vyfarbujte rovnakou farbou, nastavte teda `setLineColor("green")` aj `setFillColor("green")` alebo to isté pre farbu "black". Kružky pre priesečníky kreslite červenou farbou bez vyfarbenia, nastavte teda `setLineColor("red")` a `setNoFill()`.

**Príklad vstupu:**

```
6 8 500
250 250
400 350
250 450
50 250
250 50
350 150
200 50
100 250
100 400
300 50
300 150
300 250
300 350
300 450
```

**Príklad výstupu:**

```
nie 2
ano 1
nie 0
nie 4
ano 3
nie 2
ano 1
nie 0
```

**Obrázok body.svg:**

