

1. V Mathematici lahko točko v ravnini predstavimo s parom dveh koordinat (npr.  $\{1, 2\}$ ). Trikotnik lahko podamo s seznamom treh točk  $A$ ,  $B$  in  $C$ , navedenih v smeri nasprotni smeri urinega kazalca. Kot primer vzami trikotnik podan v obliki:

```
trikotnik = Trikotnik[{0, 0}, {5, 1}, {7, 4}]
```

Sestavi naslednje funkcije:

- `Stranice[Trikotnik[AA_, BB_, CC_]]`, ki vrne seznam stranic  $a$ ,  $b$  in  $c$  trikotnika, pri čemer `Trikotnik[AA_, BB_, CC_]` predstavlja trikotnik. Stranice so daljice, ki jih podamo kot par (seznam) dveh točk. Primer: `Daljica[{0, 0}, {5, 1}]` predstavlja stranico  $c$  za primer trikotnika podanega zgoraj. Pozor: podvojena imena točk uporabljamo, ker je spremenljivka (ime)  $C$  v Mathematici že zasedena.
- `Koti[Trikotnik[AA_, BB_, CC_]]`, ki vrne seznam kotov  $\alpha$ ,  $\beta$  in  $\gamma$  trikotnika, pri čemer kot podamo kot zaporedje treh točk. Npr. kot  $\alpha$  določajo točke  $C$ ,  $A$ ,  $B$  v tem zaporedju. Za zgornji primer bi to predstavili kot `Kot[{7, 4}, {0, 0}, {5, 1}]`. Točke naj bodo vedno navedene v smeri nasprotni urinem kazalcu (t.i. pozitivna orientacija).
- `SlikaOglisc[trikotnik_Trikotnik]`, ki vrne seznam točk kot grafičnih objektov (`Point`), ki ga lahko uporabimo za izris, če ga umestimo v objekt `Graphics`. Uporabi funkcijo `Map`. Točke naj bodo nekoliko večje.
- `SlikaStranic[trikotnik_Trikotnik]`, ki vrne seznam povezav kot grafičnih objektov (`Line`). Črte naj bodo sive barve.
- `NarisiTrikotnik[trikotnik_Trikotnik]`, ki nariše trikotnik z oglišči in stranicami. Uporabi zgornji funkciji. Izris naj bo v pravem razmerju (`AspectRatio`).

2. Sestavi naslednje funkcije

- `Dolzina[Daljica[AA_, BB_]]`, ki vrne dožino daljice med točkama  $A$  in  $B$ .
- `VelikostKota[Kot[AA_, BB_, CC_]]`, ki izračuna velikost kota v radianih.
- `VektorSimetralkota[Kot[AA_, BB_, CC_], dol_]`, ki vrne vektor nosilec simetralkota, ki je usmerjen v notranjost kota in je dolžine podane z argumentom `dol`. Namig: par  $\{a, b\}$  lahko normaliziraš z ukazom `Normalize`
- `PresecisceSimetralkota[Kot[AA_, BB_, CC_]]`, ki izračuna presečišče simetralkota z daljico med točkama  $AA$  in  $BB$ . Nastavi ustrezen sistem vektorskih enačb in poišči rešitev sistema. Točko v vrni v obliki vektorja  $\{x, y\}$ .
- `SlikaSimetralKotov[trikotnik_Trikotnik]`, ki vrne seznam slik daljic, ki predstavljajo simetralkotev. Vsaka daljica sega od kota, do presečišča s pripadajočo stranico. Nariši sliko trikotnika skupaj s simetralami kotov.
- `PresecisceSimetral[trikotnik_Trikotnik]`, ki izračuna presečišče vseh treh simetral trikotnika in ga vrne kot točko oblike  $\{x, y\}$ . Lahko si pomagaš z vsebovalnikom `Module`, ki omogoča pisanje zahtevnejših funkcij.
- `NarisiTrikotnikSSimetralami[trikotnik_Trikotnik]`, ki nariše trikotnik z oglišči, stranici, simetralami in presečiščem simetral. Trikotnik lepo oblikuj (večja oglišča, debelejša stranice, imena  $A$ ,  $B$  in  $C$  ob ogliščih in  $a$ ,  $b$  in  $c$  ob stranicah).

3. Sestavi funkcijo `NajblizjaTockaNaPremici[{AA_, BB_}, x_, y_]`, ki za dano točko  $\{x_, y_\}$  poišče najbližjo točko na premici, ki gre skozi točki AA in BB.
4. Sestavi še funkcijo `RadijVcrtanegaKroga[{AA_, BB_, CC_}]`. Ta je definiran kot razdalja med presečiščem simetral kotov ter poljubno stranico (oz. premico nosilko stranice). Pomagaj si s funkcijami iz prejšnjih dveh nalog.
5. Napiši funkcijo `NarisiTrikotnik[trikotnik_Trikotnik]` kot nadgradnjo funkcije `NarisiTrikotnikSSimetralami`, ki poleg vsega nariše še včrtan krog.
6. S pomočjo funkcije `Manipulate[...]` in funkcije `NarisiTrikotnik[trikotnik_]` sestavi interaktivno animacijo, kjer točko  $CC = cx, cy$  trikotnika lahko spreminjaš lahko tako, da velja  $cx \in [-7, 7]$  ter  $cy \in [3, 6]$ .