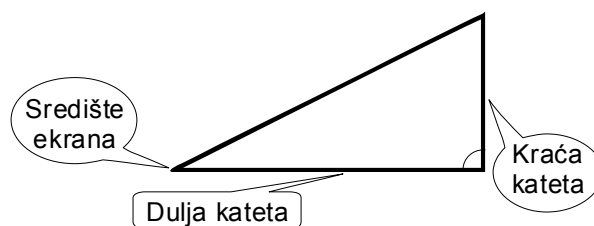


1. zadatak **TROKUT** **30 bodova**

Napišite proceduru **TROKUT :a :b** koja crta pravokutni trokut s katetama duljina :a i :b, ali tako da dulja kateta leži vodoravno. Katete su kraće stranice pravokutnog trokuta koje stoje okomito jedna na drugu.

Neka jedan vrh trokuta bude na centru ekrana.



:a je broj veći od 0, a manji od 300.

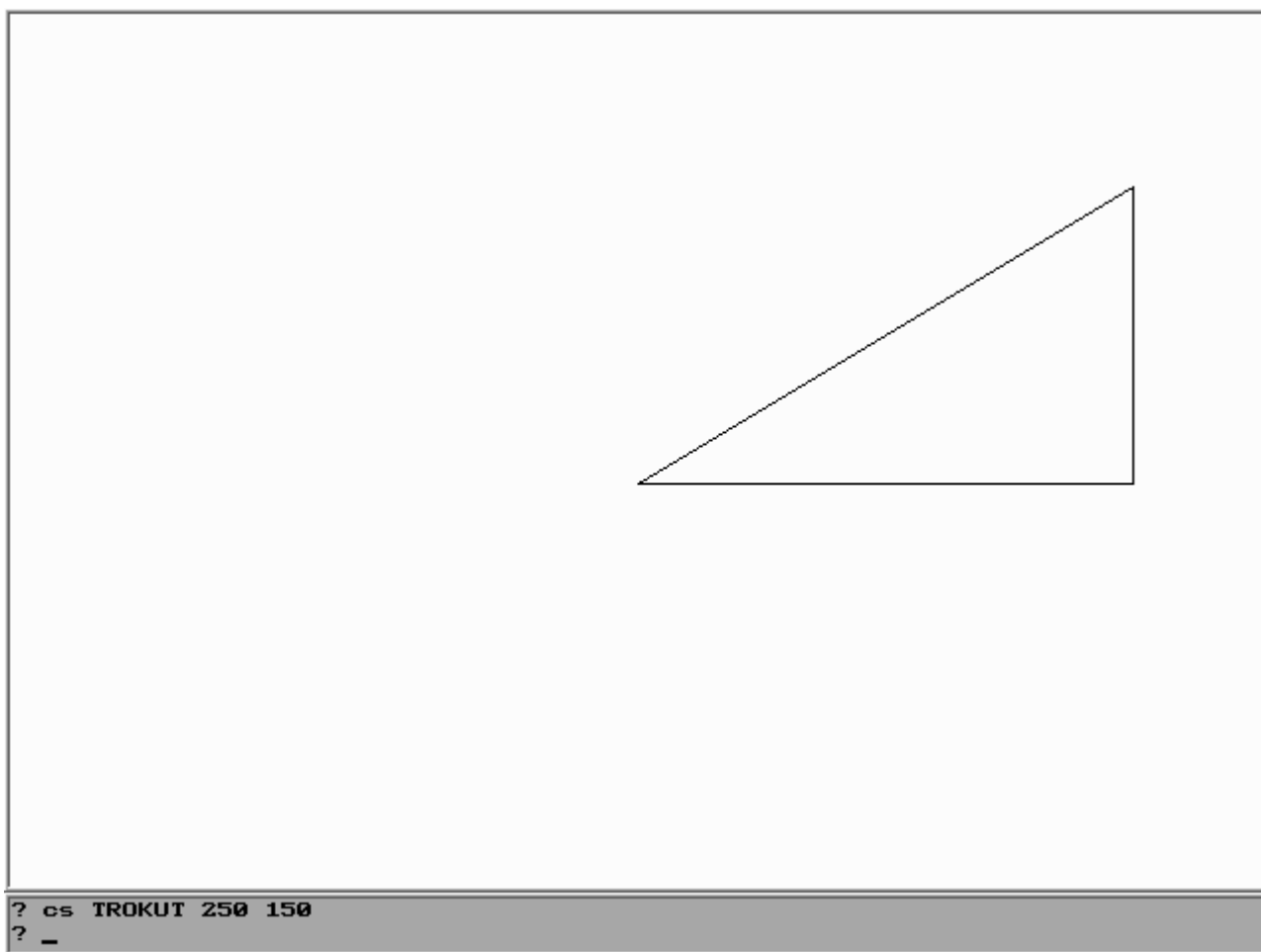
:b je broj veći od 0, a manji od 300.

Manji broj će biti uvijek manji od 200 (Tako da trokut ne prelazi rubove ekrana).

Primjeri (oba primjera vidi sliku dolje):

? cs TROKUT 250 150

? cs TROKUT 150 250



Program snimite pod imenom **TROKUT.LGO**

2. zadatak **ZVIJEZDA** **40 bodova**

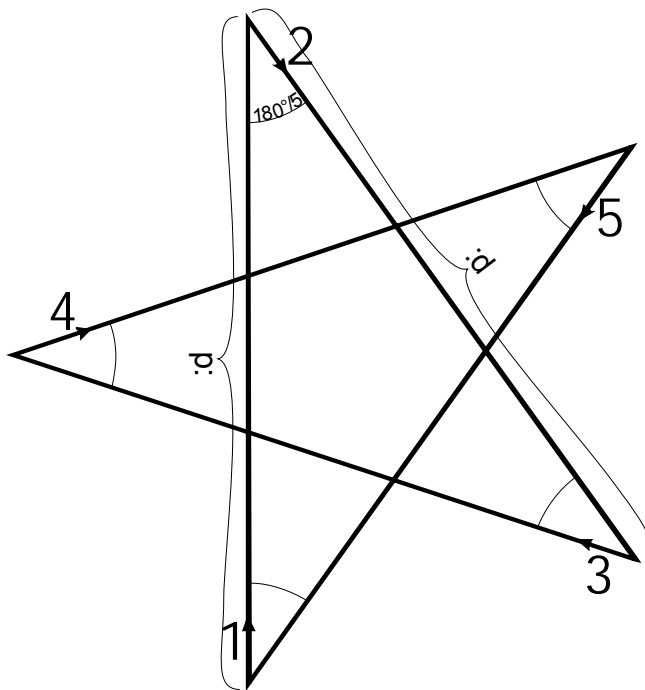
Napišite proceduru **ZVIJEZDA :n :d** koja crta zvijezdu sa **:n** krakova. Svaki vrh je spojen sa 2 nasuprotna vrha (kao na slici desno). **:d** je udaljenost nasuprotnih vrhova (kao na slici desno). Kut u vrhu svakog kraka je $\frac{180^\circ}{:n}$.

:n je neparan broj veći od 2.

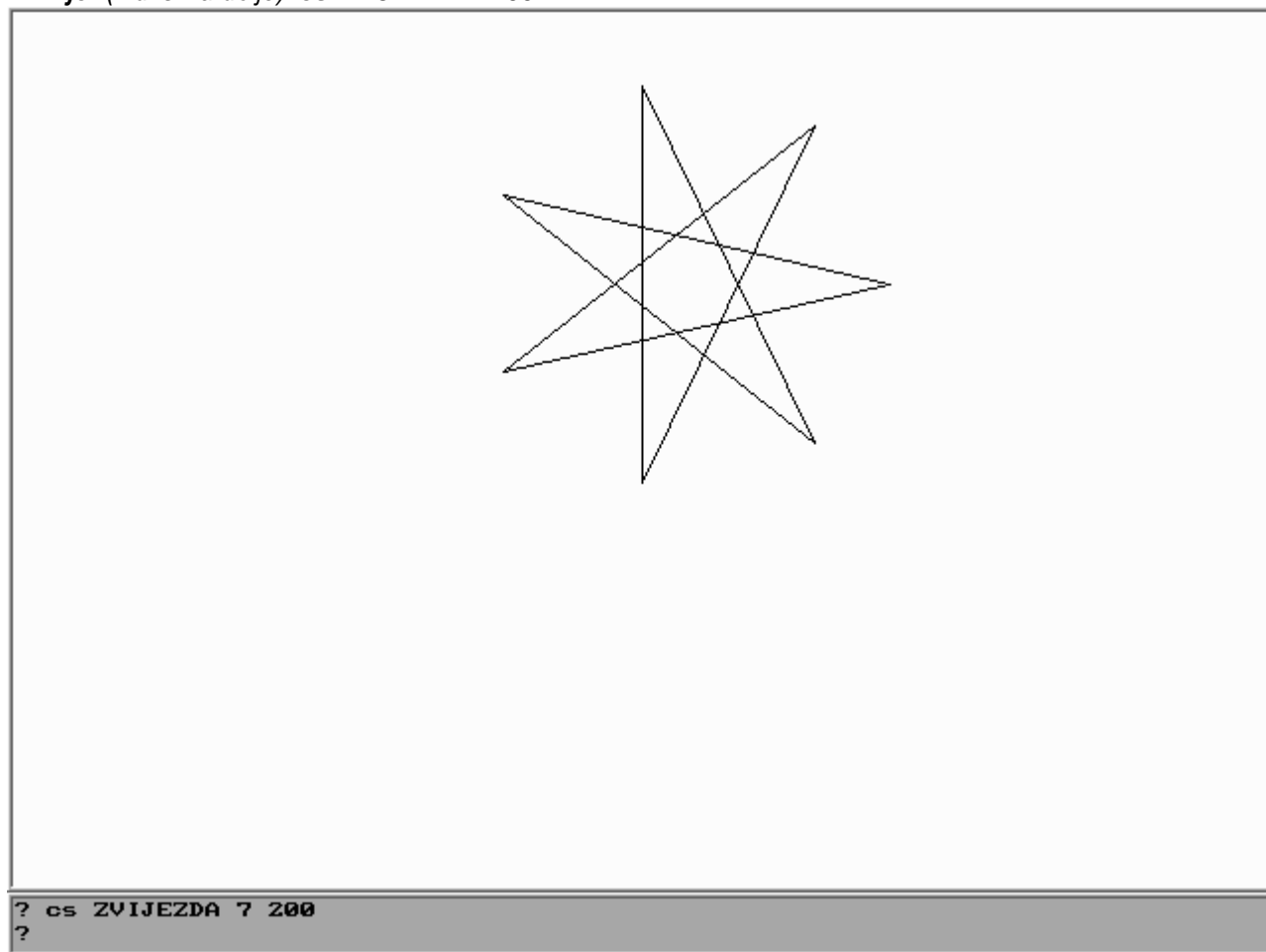
:d je broj veći od 0.

Pozicija lika na ekranu nije bitna.

Na primjer na slici desno je nacrtana zvijezda sa 5 krakova. Jedan od mogućih načina da se ona nacrtat bio bi da se crtaju linije 1, 2, 3, 4 i 5.



Primjer (vidi sliku dolje): cs ZVIJEZDA 7 200



Program snimite pod imenom **ZVIJEZDA.LGO**

3. zadatak **ENT** **60 bodova**

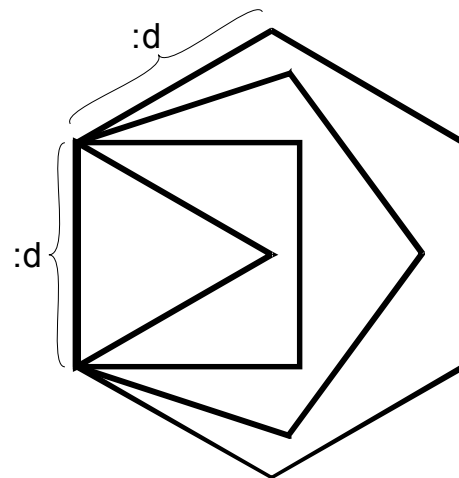
Napišite proceduru **ENT :n :d** koja crta pravilne mnogokute jedan unutar drugog, tako da im je jedna stranica svima zajednička (kao na slici desno). Svi mnogokuti imaju duljine stranica **:d**. Unutarnji mnogokut je trokut. Svaki slijedeći mnogokut ima jednu stranicu više od prethodnog, dok vanjski mnogokut ima **:n** stranica (na slici desno šesterokut).

Na slici desno je primjer za **:n=6**.

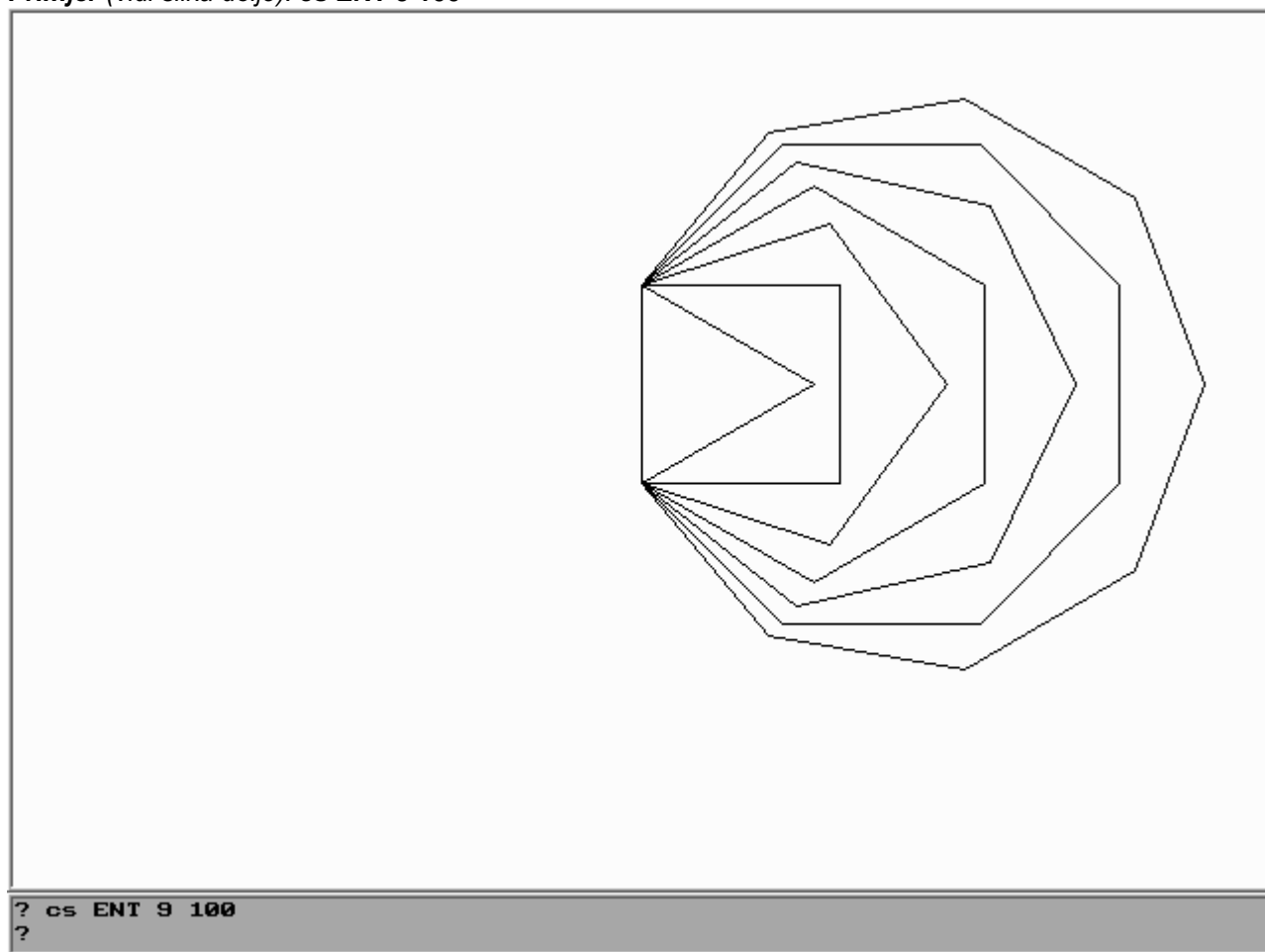
:n je prirodni broj veći od 2.

:d je broj veći od 0.

Pozicija lika na ekranu nije bitna.



Primjer (vidi sliku dolje): cs ENT 9 100



Program snimiti pod imenom **ENT.LGO**

4. zadatak**PROST****70 bodova**

Prost broj je takav prirodan broj koji je djeljiv samo sa brojem 1 i sa samim sobom. Broj 1 nije prost broj!

Npr. prosti brojevi su 2, 3, 5, 7, 11, 13, 17..., jer nisu djeljivi ni sa jednim drugim brojem (osim sa 1 i sa samim sobom).

Broj 4 nije prost, jer je djeljiv i sa 2.

Broj 6 nije prost, jer je djeljiv i sa 2 i 3.

Broj 8 nije prost, jer je djeljiv i sa 2 i 4.

Broj 9 nije prost, jer je djeljiv i sa 3.

Broj 10 nije prost, jer je djeljiv i sa 2 i 5.

Napišite funkciju (procedura koja vraća vrijednost naredbom **op**) **PROST :n** koja nalazi najmanji prost broj veći ili jednak od **:n**.

:n je prirodan broj manji od 10 000.

Savjet za PC-LOGO: Ako ste upisali velik broj **:n** i program se dugo izvršava, CTRL-C prekida izvršavanje.

Primjeri:

```
? pr PROST 7
```

```
7
```

Jer je broj 7 prost broj!

```
? show PROST 1
```

```
2
```

Jer broj 1 nije prost po definiciji, a 2 je!

```
? PROST 8
```

```
Result: 11
```

Jer brojevi 8, 9 i 10 nisu prosti, a 11 je!

```
? PROST 15
```

```
Result: 17
```

Jer brojevi 15 i 16 nisu prosti, a 17 je!

```
? PROST 140
```

```
Result: 149
```

Jer brojevi 140..148 nisu prosti, a 149 je!

```
? PROST 9999
```

```
Result: 10007
```

Jer brojevi 9 999..10 006 nisu prosti, a 10 007 je!
(Možda će programu trebati par sekundi da se ovo izračuna)

Program snimiti pod imenom **PROST.LGO**