

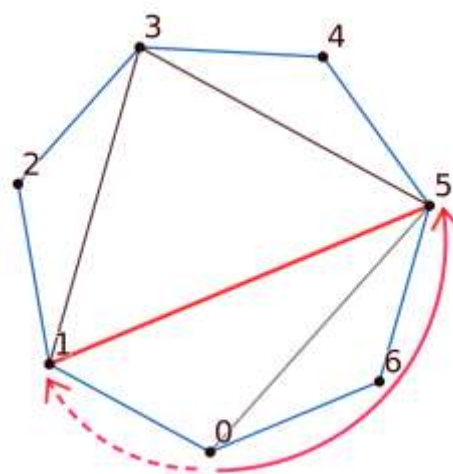
Triangulatsioon

Ülesanne

Anna joonistas korrapärase n -nurga ja nummerdas selle tipud päripäeva 0 kuni $n - 1$. Seejärel jagas ta hulknurga kolmnurkadeks, joonistades $n - 3$ üksteist mittelõikavat diagonaali. Diagonaalid võivad siiski üksteist puutuda nende ühistes otspunktides. Iga diagonaal on sirglõik kahe erineva tipu vahel, millel pole ühist serva.

Kauguse tipu A ja diagonaali D vahel defineerime järgmiselt. Alustame tipust A ja liigume edasi päripäeva kuni jõuame ühte diagonaali D otspunkidest. Nimetame selle käigus läbitud servade arvu **vasakpoolseks** kauguseks. Samamoodi nimetame **parempoolseks** kauguseks servade arvu, mille läbime diagonaali D otspunkti jõudmiseks tipust A alustades ja vastupäeva liikudes. Tipu A ja diagonaali D vaheliseks **kauguseks** nimetame vasakpoolse ja parempoolse kauguse **maksimumi**.

Näiteks on joonisel tipu 0 ja diagonaali (1,5) vaheline kaugus 2, sest vasakpoolne kaugus on 1 ja parempoolne 2. Diagonaali (0,5) kaugus tipust 0 on 5, sest vasakpoolne kaugus on 5 ja parempoolne 2.



Anna tahab Jacobile mõistatuse esitada. Jacob ei tea, millised diagonaalid Anna joonistas. Ta teab ainult n väärtust ja võib Anna käest küsida, kas mingi tippude paari vahel on diagonaal või mitte. Jacobi ülesanne on leida tipule 0 lähim diagonaal (diagonaali kaugust tipust arvestame, nagu eelpool defineeritud). Aita Jacobil see ülesanne lahendada, kasutades piiratud arvu küsimusi.

Piirangud

- $5 \leq n \leq 100$

Suhtlus

Sinu ülesanne on realiseerida funktsioon

```
int solve(int n)
```

- Hindaja kutsub seda funktsiooni välja täpselt ühe korra
- n on hulknurga tippude arv
- Funktsioon peab tagastama diagonaal tippude a ja b vahel kui täisarvu $a \cdot n + b$
- Kui minimaalse kaugusega diagonaale on mitu, võib tagastada ükskõik millise neist

Funktsioon **solve** võib oma töö käigus kasutada funktsiooni

```
int query(int x, int y)
```

- x on esimese tipu number
- y on teise tipu number
- $0 \leq x, y \leq n - 1$
- Funktsioon tagastab 1, kui tippude x ja y vahel on diagonaal, ja 0, kui ei ole

Suhtluse näide

Siin on hindaja sisendi ja erinevate funktsioonide vaheliste kutsete näide. Näide vastab eelpool olevale joonisele.

Hindaja sisendi ainsal real on täisarv n .

Näidishindaja väljastab iga funktsiooni **query** kutsele vastava küsimuse standardväljundisse ja sa pead ise sisestama vastuse 0 või 1.

Hindaja sisend	Funktsioonide väljakutsed			
	Kutse	Tulemus	Kutse	Tulemus
7	solve(7)			
			query(0, 3)	
				query tagastab 0
			query(0, 5)	
				query tagastab 1
			query(1, 5)	
				query tagastab 1
		solve tagastab $1 \cdot 7 + 5 = 12$		
		Õige vastus!		

Hindamine

Olgu q sinu programmi ühes testis esitatud küsimuste arv ja $w = \frac{n \cdot (n-3)}{2}$.

- Kui sinu programm esitab vigase küsimuse või annab vale vastuse, saad sa 0% selle testi punktidest
- Kui $w < q$, saad sa 0% selle testi punktidest
- Kui $n < q \leq w$, saad sa $10 + 60 \cdot \frac{w-q}{w-n}\%$ selle testi punktidest
- Kui $q \leq n$, saad sa 100% selle testi punktidest

Alamülesanded

Selles ülesandes on ainult üks alamülesanne ja punkte saab iga testi eest eraldi.

Võistluse ajal saad sa tagasiside ainult poolte testide kohta (koguväärtusega 50 punkti).

Teise poole tulemused avaldatakse alles pärast võistluse lõppu. **Hindamisel läheb arvesse parim esitatud lahendus.**