

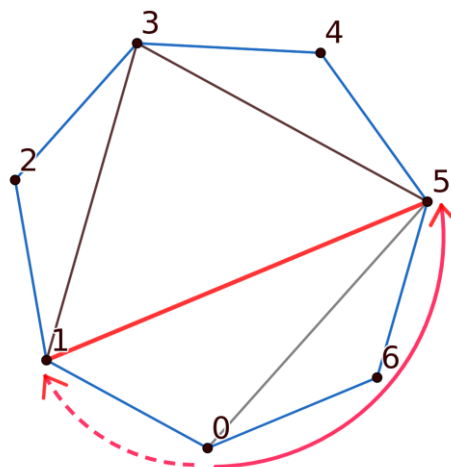
# Триангулация

## Условие

Анна начерта правилен многоъгълник с  $n$  върха, номерирани от 0 до  $n - 1$  по посока на часовниковата стрелка. По-късно тя раздели многоъгълника на триъгълници, като начерта  $n - 3$  диагонала, които не се пресичат един с друг, с изключение на допиране в крайните точки. Диагонал се дефинира като права линия между два различни върха, които не образуват страна на многоъгълника.

Нека дефинираме разстояние между връх  $A$  до диагонал  $D$ . Да предположим, че започваме от връх  $A$  и се движим по посока на часовниковата стрелка, докато не достигнем крайна точка на диагонала  $D$ . Броят на изминатите страни ще наричаме **ляво\_разстояние**. Аналогично, **дясно\_разстояние** е броят изминати страни, ако почнем от  $A$  и се движим по посока обратна на часовниковата стрелка докато не достигнем другата крайна точка на  $D$ . **Разстоянието** от  $A$  до  $D$  е **максимумът** от **ляво\_разстояние** и **дясно\_разстояние**.

На примерното изображение разстоянието от връх 0 до диагонала  $(1,5)$  е 2, като ляво\_разстояние е 1, а дясно\_разстояние е 2. А за диагонала  $(0,5)$  разстоянието от връх 0 е 5, като ляво\_разстояние=5 и дясно\_разстояние=2.



Анна иска да предизвика Яков. Той не знае кои диагонали са начертани. Единствено знае стойността на  $n$ , но може да пита Анна неколkokратно за двойка върхове и тя ще му каже дали има диагонал между тези върхове. Целта на Яков е да намери най-близкия (съгласно горната дефиниция на разстояние) диагонал спрямо връх 0. Вие трябва да му помогнете да постигне целта си със задаването на ограничен брой въпроси към Анна.

## Ограничения

- $5 \leq n \leq 100$

## Детайли по реализацията

Вие трябва да напишете следната функция във вашето решение:

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

- Тази функция се вика точно веднъж от грейдъра
- $n$ : броят върхове в многоъгълника
- Тази функция трябва да върне диагонала между два върха  $a$  и  $b$  като цяло число със стойност  $a \cdot n + b$
- Ако има няколко възможни диагонала с минимално разстояние, може да върнете който и да е от тях

Горната функция може да извиква следната функция:

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

- $x$ : номерът на единия връх
- $y$ : номерът на другия връх
- $0 \leq x, y \leq n - 1$
- връща 1, ако има диагонал между  $x$  и  $y$  и 0 в противен случай

## Примерна комуникация

Това е примерен вход за грейдъра и съответните извиквания на функции. Този вход е показан на горното изображение.

Единственият ред на входа съдържа едно цяло число:  $n$

Примереният грейдър ще отпечата всяко извикване на `query` на стандартния изход и вие ще трябва ръчно да отговорите с 1 или 0.

Примерен вход за грейдъра	Примерни извиквания			
	Извикване	Върнат резултат	Извикване	Върнат резултат
7	solve(7)			
			query(0, 3)	
				query връща 0
			query(0, 5)	
				query връща 1
			query(1, 5)	
				query връща 1
		solve връща $1 \cdot 7 + 5 = 12$		
		Correct!		

## Оценяване

Нека означим с  $q$  броят въпроси, които сте задали на даден тест. Допълнително,  $w = \frac{n \cdot (n-3)}{2}$ .

- Ако питате невалиден въпрос или върнете неправилен отговор, ще получите 0% от точките за теста
- Ако  $w < q$ , то ще получите 0% от точките за теста
- Ако  $n < q \leq w$ , то ще получите  $10 + 60 \cdot \frac{w-q}{w-n}\%$  от точките за теста
- Ако  $q \leq n$ , то ще получите 100% от точките за теста

## Подзадачи

Има единствена подзадача, като броят точки ще е сумата от броя точки на тестовите по отделно. Но по време на състезанието ще имате възможност да виждате броя точки само на половината тестове (за общо 50 точки). Точките за другата половина тестове ще бъдат видими след края на състезанието. Вашият краен резултат ще е **най-добрият резултат сред всички пратени решения по задачата**.