

# Картковий фокус

Двоє гравців збираються продемонструвати картковий фокус із стандартною колодою на 52 карти. Для зручності значення карт будуть різними цілими числами від 0 до 51.

Кarti спочатку кладуть на стіл в один рядок лицьовою стороною догори (з видимими значеннями) в якомусь порядку, невідомому гравцям.

Перший гравець підходить до столу, дивиться на карти і робить обмін, щонайбільше  $S$  разів. Кожен обмін виконується шляхом вибору двох карт на позиціях  $i$  та  $j$  ( $i$  та  $j$  можуть бути рівними) і обміну цих карт місцями.

Після цього перший гравець відходить, не спілкуючись з другим гравцем, і всі карти перевертаються (їх значення більше не видно), не змінюючи порядок. Другого гравця запрошують до столу і просять вгадати, де знаходиться карта з **цільовим** значенням, і йому дозволяється перевернути щонайбільше  $T$  карт. Якщо будь-яка з розкритих карт є **цільовою**, тоді гравці виграють. Якщо у них закінчуються спроби, вони програють.

Ваша мета - написати дві програми, які будуть імітувати дії гравців та вигравати гру.

## Деталі реалізації

Вам буде надано дві програми - FirstPlayer і SecondPlayer разом із зразком грейдера.

У FirstPlayer вам треба реалізувати таку функцію:

```
void swapCards(int cards[], int S, int T)
```

- Ця функція буде викликана рівно один раз грейдером
- cards: масив, що містить початкові значення карт зліва направо, з рівно 52 елементами, проіндексованими від 0 до 51
- S: кількість допустимих обмінів
- T: кількість спроб перевернути карту

**swapCards** може здійснювати виклики до наступної функції:

```
void doSwap(int i, int j)
```

- $i$ : індекс першої карти для обміну,  $0 \leq i < 52$
- $j$ : індекс другої карти для обміну,  $0 \leq j < 52$
- **doSwap** може бути викликана не більше S разів

У SecondPlayer вам треба реалізувати таку функцію:

```
void guessCard(int S, int T, int target)
```

- S: кількість допустимих обмінів
- T: кількість спроб перевернути карту
- target: вартість картки, яку слід розкрити

**guessCard** може здійснювати виклики до наступної функції:

```
int guess(int idx)
```

- idx: індекс, що перевіряється,  $0 \leq idx < 52$
- Функція повертає значення карти на позиції idx
- **guess** може бути викликана не більше T разів
- коли цільова карта розкривається, тестування успішно закінчується

## Приклад взаємодії

Нижче наведено приклад вхідних даних для грейдера.

Перший рядок повинен містити два цілих числа: S і T.

Другий рядок повинен містити 52 числа. i-те число позначає вартість i-ої карти.

Третій рядок містить одне ціле число – **цільову** карту.

Приклад вхідних даних для грейдера	Прикладові виклики функцій		
	Виклики	Підвиклики	Результат
1 51 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 1	swapCards([0,1,...], 1, 51)		
		doSwap(0, 1)	
			мінняє місцями карти на позиціях 0 та 1
			swapCards завершується
	guessCard(1, 51, 1)		
		guess(5)	
			guess повертає 5
		guess(1)	
			guess повертає 0
		guess(0)	
			Все вірно

## Обмеження

- $1 \leq S \leq 52$
- $1 \leq T \leq 51$
- $0 \leq target < 52$ , де *target* – це вартість цільової карти

## Блоки

1. (16 балів):  $S = 52, T = 1$
2. (20 балів):  $S + T = 52$
3. (22 балів):  $S = 13, T = 27$
4. (18 балів):  $S = 1, T = 26$
5. (24 балів): Виграшна стратегія існує для даних  $S$  і  $T$ .