

Card Trick

Doi jucători doresc să realizeze un truc folosind un pachet standard cu 52 de cărți de joc. Pentru comoditate, valorile cărților sunt distincte, cuprinse între 0 și 51.

Inițial cărțile sunt așezate pe o masă, formând un singur rând, cu fața în sus (cu valorile vizibile), într-o anumită ordine necunoscută de cei doi.

Primul jucător merge la masă, privește cărțile și realizează cel mult S interschimbări.

Fiecare interschimbare constă în alegerea a două cărți de pe pozițiile i și j (i și j pot fi egale) și duce cartea de pe poziția i în locul celei de pe poziția j , și invers.

După aceasta primul jucător pleacă fără a comunica cu primul și toate cărțile vor fi întoarse (deci nu se mai vede ce conțin), fără să li se schimbe ordinea. Al doilea jucător este invitat la masă și este întrebat unde se găsește o anumită carte cu o anume valoare **țintă** permițându-i-se să întoarcă cel mult T cărți, una câte una. Dacă una dintre cărțile întoarse este cea **țintă**, atunci cei doi jucători câștigă. Scopul tău este să scrii două programe care să simuleze acțiunile celor doi și să câștige jocul.

Implementation Details

Vi se vor furniza două programe - FirstPlayer și SecondPlayer, împreună cu un exemplu de evaluator.

În FirstPlayer ai de implementat următoarea funcție:

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

- Funcția este apelată exact o dată de către evaluator
- cards: un tablou conținând inițial valorile de la stanga la dreapta, cu exact 52 de elemente, indexat de la 0 la 51
- S: numărul de interschimbări
- T: numărul de ghiciri permise

swapCards poate apela următoarea funcție:

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

- i : indicele primei cărți de interschimbat, $0 \leq i < 52$
- j : celei de-a doua cărți de interschimbat, $0 \leq j < 52$
- doSwap poate fi apelată de cel mult S ori

În SecondPlayer ai de implementat funcția următoare:

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

- S: numărul de interschimbări
- T: number de ghiciri permise
- target: cartea a cărei valori trebuie descoperită

guessCard poate apela funcția următoare:

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

- idx: indicele ghicit, $0 \leq idx < 52$
- Returnează valoarea cărții de indicele i.
- guess poate fi apelată de cel mult T ori.
- când apare ghicirea, evaluarea se termină cu succes

Sample interaction

Mai jos este un exemplu de input pentru evaluatorul atașat.

Prima linie trebuie să conțină doi întregi: S și T.

A doua linie trebuie să conțină 52 de numere. Al i-lea reprezintă valoarea cărții i.

Al treilea conține un întreg **ținta**.

Exemplu de intrare pentru evaluator	Exemple de apeluri		
	apeluri	subapeluri	Returnări
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)	
			Interschimbă cărțile de la indicii 0 și 1.
			swapCards finishes.
	guessCard(1, 51, 1)		
		guess(5)	
			guess returns 5

		guess(1)	
			guess returns 0
		guess(0)	
			Correct!

Constraints

- $1 \leq S \leq 52$
- $1 \leq T \leq 51$
- $0 \leq \text{ținta} < 52$

Subtasks

1. (16 puncte): $S = 52, T = 1$
2. (20 puncte): $S + T = 52$
3. (22 puncte): $S = 13, T = 27$
4. (18 puncte): $S = 1, T = 26$
5. (24 puncte): Există strategie de câștig pentru S and T