

ფოკუსი კარტით

ორი მოთამაშე აპირებენ გიჩვენონ ფოკუსი სტანდარტული 52-კარტიანი დასტით. სიმარტივისთვის კარტები გადავწვამოთ 0-დან 51-ის ჩათვლით. თავიდან კარტები დალაგებულია მაგიდაზე ერთ რიგად (გამოჩენილი მნიშვნელობებით), მოთამაშეებისთვის უცნობი თანმიმდევრობით. პირველი მოთამაშე მიდის მაგიდასთან, დახედავს კარტების მიმდევრობას და გააკეთებს მაქსიმუმ **S** გაცვლას. თითოეული გაცვლა გულისხმობს ორი კარტის ადგილმდებარეობების (i და j) არჩევას და მათი ადგილების გაცვლას. შემდეგ პირველი მოთამაშე ტოვებს მაგიდას მეორე მოთამაშესთან კომუნიკაციის გარეშე. შემდეგ ყველა კარტი ამობრუნდება თანმიმდევრობის შეცვლის გარეშე (მნიშვნელობები აღარ ჩანს). მეორე მოთამაშე მოდის მაგიდასთან ჩაფიქრებული კარტის (**target**) მოსაძებნად. მას აქვს უფლება ამოატრიალოს მაქსიმუმ **T** კარტი რიგრიგობით. თუ ამოატრიალებული კარტებიდან ერთ-ერთი აღმოჩდება ჩაფიქრებული კარტი (**target**), მოთამაშეები იგებენ თამაშს. წინააღმდეგ შემთხვევაში აგებენ. შენი მიზანია დაწერო ორი პროგრამა რომელებიც იმოქმედებენ მოთამაშეების მაგივრად ისე, რომ მოთამაშეებმა მოიგონ თამაში.

განხორციელების დეტალები

მოცემული გეგნებით ორი პროგრამა FirstPlayer და SecondPlayer თანდართული სამაგალითო გრადერით.

FirstPlayer-ში უნდა დაწეროთ შემდეგი ფუნქცია:

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

- ფუნქცია გამოძახებული იქნება მხოლოდ ერთხელ.
- cards: შეიცავს 52 ელემენტიან მთელ რიცხვთა მასივს - კარტების საწყისი განლაგება მარცხნიდან მარჯვნივ, თითოეული რიცხვი არის [0, 51] ინტერვალში.
- S: დაშვებული გაცვლების რაოდენობა.
- T: დაშვებული გამოცნობების რაოდენობა.

swapCards-დან შესაძლოა გამოიძახოთ შემდეგი ფუნქცია:

void doSwap(int i, int j)

- i: პირველი გასაცვლელი კარტის ინდექსი, $0 \leq i < 52$
- j: მეორე გასაცვლელი კარტის ინდექსი, $0 \leq j < 52$
- **doSwap** შესაძლოა გამოიძახოთ მაქსიმუმ S-ჯერ.

SecondPlayer-ში უნდა დაწეროთ შემდეგი ფუნქცია:

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

- S: დაშვებული გაცვლების რაოდენობა.
- T: დაშვებული გამოცნობების რაოდენობა.
- target: ჩაფიქრებული კარტის მნიშვნელობა.

guessCard-დან შესაძლოა გამოიძახოთ შემდეგი ფუნქცია:

int guess(int idx)

- idx: ამოსატრიალებელი კარტის ინდექსი, $0 \leq idx < 52$
- აბრუნებს კარტის მნიშვნელობას idx ინდექსზე.
- **guess** შესაძლოა გამოიძახოთ მაქსიმუმ T-ჯერ.
- საძიებო კარტის აღმოცენის შემთხვევაში შეფასება სრულდება წარმატებით.

მაგალითი კომუნიკაცია

ქვემოთ მოცემულია შესატანი მონაცემების მაგალითი თანდართულ გრაფერის გამოყენებით.

პირველ ხაზზე შემოდის ორი რიცხვი S და T.

მეორე ხაზზე შემოდის 52 რიცხვი. i-ური რიცხვი არის მარცხნიდან i-ური კარტის მნიშვნელობა მაგიდაზე.

მესამე ხაზზე შემოდის ჩაფიქრებული კარტის მნიშვნელობა (**target**).

შესატანი მონაცემები გრადერს	მაგალითი გამოძახებები		
	გამოძახებები	ქვე- გამოძახებები	დაბრუნებები
1 51	swapCards([0,1,...], 1, 51)		
0 1 2 3 4 5 6 7 8		doSwap(0, 1)	
9 10 11 12 13			ცვლის კარტებს ინდექსებზე 0 და 1
14 15 16 17 18	swapCards finishes		.
19 20 21 22 23	guessCard(1, 51, 1)		
24 25 26 27 28		guess(5)	
29 30 31 32 33			guess აბრუნებს 5
34 35 36 37 38		guess(1)	
39 40 41 42 43			guess აბრუნებს 0
44 45 46 47 48		guess(0)	
49 50 51			სწორია!
1			

შეზღუდვები

- $1 \leq S \leq 52$
- $1 \leq T \leq 51$
- $0 \leq target < 52$

ქვეამოცანები

1. (16 points): $S = 52, T = 1$
2. (20 points): $S + T = 52$
3. (22 points): $S = 13, T = 27$
4. (18 points): $S = 1, T = 26$
5. (24 points): გარანტირებულია მომგებიანი სტრატეგიის არსებობა მოცემული S და T -სთვის.