



## KARAWANY

Planarystan – jedyny kraj z siecią dróg będącą grafem planarnym – słynie z tego, że jest kolebką cywilizacji. Każde miasto może się pochwalić ogromnym kompleksem ruin, a najważniejszym towarem Planarystanu są antyki. Co więcej, antyki z każdego miasta są unikalne. Archeolog Archibald widział katastrofalne skutki prowadzenia gospodarki opartej na zabytkach i postanowił zaprotestować. Ponieważ okupacja stanowisk archeologicznych nie przyniosła efektu, postanowił złożyć projekt ustawy. Rząd przegłosował ustawę zakazującą handlu antykami wewnątrz kraju i uznającą handlarzy za przestępców. Ustawa wejdzie w życie za **T** dni. Jako że jesteś handlarzem antyków, postanawiasz poświęcić ten czas na zgromadzenie jak największej sumy pieniędzy na swoim koncju na Kajmanach. Masz zarezerwowany wylot 10 minut przed momentem wejścia ustawy w życie. Pozostali handlarze wpadli na ten sam pomysł.

### Opis rozgrywki

Planarystan można opisać jako graf planarny. Przemysłni inżynierowie zbudowali sieć drogową tak, że odległości między miastami są zawsze liczbami naturalnymi. Handel antykami rządzi się paroma prawami (i sporą porcją bezprawia). Każdy handlarz może mieć maksymalnie **C** karawan przewożących antyki. Karawany mają ładowność i prędkość poruszania się opisane liczbami naturalnymi. Ze względu na problemy ze smogiem ładowność i prędkość karawany muszą sumować się do **K**. Handlarz, który jeszcze nie ma maksymalnej liczby karawan, może wprowadzić do obiegu karawanę w dowolnym mieście z bankiem. Świeżo dodana karawana będzie mogła wyjechać z miasta następnego dnia. Karawana zawsze przemieszcza się o maksymalną możliwą odległość w zadanym kierunku. Jeśli na początku dnia karawana znajduje się na drodze, to przemieszcza się automatycznie w kierunku celu. Z raz obranej drogi nie można zawrócić. Karawana po wjeździe do miasta traci pozostałe jej punkty ruchu w korkach, tj. może opuścić miasto dopiero kolejnego dnia.

Karawana może w dowolnym mieście kupić pewną liczbę naturalną sztuk antyków. Jedna sztuka zawsze kosztuje **P**. Za antyki należy płacić gotówką. Karawana może wypłacić część pieniędzy z konta handlarza w dowolnym mieście z bankiem. Oczywiście w tych samych warunkach możliwe są wpłaty na konto. Antyki można sprzedać w dowolnym mieście. Cena sprzedaży to  $P + D^2$ , gdzie **D** to długość najkrótszej ścieżki między miastem pochodzenia antyku a miastem sprzedaży.

Handel antykami przynosi kolosalne zyski, nie dziwota więc, że swój udział w nim chcieliby mieć burmistrzowie miast. Dlatego każdy z nich przyjmuje łapówki od handlarzy. Na koniec miesiąca (długość miesiąca to **M**) każdy z nich podlicza przelewy i temu handlarzowi, który w tym miesiącu przelał najwięcej, zapewnia preferencyjne warunki w mieście. Jeśli istnieje więcej niż jeden handlarz, który wpłacił największą kwotę, wtedy żaden nie dostaje lepszych warunków.

Preferencyjne warunki działają w dwojaki sposób. Jeśli handlarz, który wpłacił najwięcej, wpłacił też co najmniej 10% całkowitej kwoty łapówek dla tego miasta, to przez najbliższy miesiąc karawany innych graczy nie mogą wyjechać z miasta. Jeśli zaś najwyższa wpłata przekroczyła 50% całkowitej kwoty łapówek, to dodatkowo karawanom innych graczy wjeżdżającym do tego miasta odbierana jest gotówka i towary (które są na miejscu sprzedawane). Zdobyta w ten sposób kwota handlarza, który wpłacił najwięcej. Jeżeli żaden z powyższych warunków nie został spełniony, miasto nie zostaje objęte żadną blokadą. Łapówki nie działają na zasadzie licytacji, tj. wpłaty nie podlegają zwrotowi niezależnie od wyniku rozliczenia łapówki. Na wynik rozliczenia łapówek pod koniec miesiąca nie wpływają łapówki wpłacone w miesiącach poprzednich.

Burmistrzowie miast, w których jest bank, nie uczestniczą w przekupstwach związanych z antykami.

### Opis ruchów

Rozgrywka jest podzielona na tury (tura odpowiada jednemu dniowi). W trakcie tury gracz może wydać każdej karawanie jedno polecenie ruchu, a poleceń łącznie – 50. Pod koniec tury serwer wykonuje wszystkie polecenia w kolejności zgłoszenia. Jeśli liczba tur pozostałych do końca dzieli się przez **M**, to na samym początku nowej tury burmistrzowie zliczają łapówki. Karawany znajdujące się w drodze między miastami automatycznie dostają polecenie ruchu. Handlarze mają dostęp do stanu mapy z początku tury (po rozliczeniu łapówek, ale przed wykonaniem dowolnych ruchów w tej turze).

## Opis serwera gry

Gra jest podzielona na rundy – runda odpowiada czasowi pozostałemu do wejścia ustawy w życie. Rundy z kolei dzielą się na tury. W każdej rundzie wszystkie drużyny konkurują ze sobą na jednej mapie. Punkty są podliczane na koniec każdej rundy. Ranking jest ustalany według wzoru:

$$\text{SCORE}(id) = \text{MONEY}(id) + \sum_{c \in \text{CARAVANS}(id)} \lfloor \frac{\text{MONEY}(c)}{10} \rfloor$$

Zmienna *id* to identyfikator gracza, a *MONEY(id)* to stan konta gracza o identyfikatorze *id*.

## Dokładny opis poleceń

Każde polecenie – jeśli było poprawne – zwraca natychmiast OK. W przeciwnym wypadku zwraca komunikat ERR z numerem błędu i krótkim opisem.

### GET\_TURNS\_LEFT

Zwraca liczbę tur pozostałych do końca rundy. Przykład:

```
> GET_TURNS_LEFT
< OK
< 28
```

### WAIT

Natychmiast zwraca OK (jak każda inna komenda), po czym zwraca dodatkowe OK NEXT\_TURN na początku nowej tury albo OK NEXT\_ROUND na początku nowej rundy. Przykład:

```
> WAIT
< OK
(po pewnym czasie...)
< OK NEXT_TURN
```

### GET\_SCOREBOARD

Zwraca **U** – liczbę graczy. Następnie zwraca **U** zestawów danych **u s** – odpowiednio identyfikatory graczy i ich obecna liczba punktów. Przykład:

```
> GET_SCOREBOARD
< OK
< 2
< 937 15435
< 4072 65532
```

### GET\_CONSTANTS

Zwraca cztery liczby – **K C P M** – wspomniane w opisie gry (odpowiednio: ograniczenie smogowe, liczba karawan, cena jednego antyku, długość miesiąca w dniach). Przykład:

```
> GET_CONSTANTS
< OK
< 12 2 1000 15
```

### GET\_MAP

Zwraca dwie liczby – **V E**. **V** oznacza liczbę miast w Planarystanie, a **E** – liczbę dróg. Kolejno następuje **E** zestawów danych **id1 id2 l** – identyfikatory miast połączonych drogą i jej długość. Identyfikatory miast to liczby naturalne od 0 do **V** – 1. Przykład:

```
> GET_MAP
< OK
< 3 3
```

```
< 0 1 2
< 1 2 2
< 2 0 2
```

## GET\_BANKS

Zwraca liczbę **B** oznaczającą liczbę banków w Planarystanie. Kolejne **B** liczb to identyfikatory miast, w których jest bank. Zakres wartości identyfikatorów miast podany jest przy opisie **GET\_MAP**. Przykład:

```
> GET_BANKS
< OK
< 2
< 0
< 2
```

## MY\_ID

Zwraca liczbę będącą identyfikatorem waszego handlarza. Przykład:

```
> MY_ID
< OK
< 653
```

## GET\_CARAVANS

Zwraca **c** – liczbę karawan działających w Planarystanie. Potem następuje **c** zestawów danych **pid cid id1 d id2 l m s**. Wartość **pid** to identyfikator właściciela karawany, a wartość **cid** to identyfikator karawany. **id1 d id2** opisuje pozycję karawany. Jeśli **d** = 0, to **id1** = **id2** i karawana znajduje się w mieście o numerze **id1**. W przeciwnym przypadku karawana znajduje się na krawędzi z miasta o identyfikatorze **id1** do miasta o identyfikatorze **id2** i jest w odległości **d** od miasta **id2**. **l** to liczba antyków transportowanych przez karawanę, a **m** to jej maksymalna pojemność. Prędkość ruchu karawany to **s**. Przykład:

```
> GET_CARAVANS
< OK
< 2
< 7383 435 1 0 1 5 6 7
< 7383 9856 5 8 4 2 2 11
```

## ADD\_CARAVAN

Przyjmuje trzy liczby – **l s t**. Liczby **l** i **s** oznaczają kolejno maksymalną pojemność i prędkość ruchu karawany. **t** to identyfikator miasta, w którym karawana ma się pojawić. Zwracana jest liczba – identyfikator właśnie dodanej karawany. Przykład:

```
> ADD_CARAVAN 6 7 2
< OK
< 65
```

## MY\_CARAVANS

Działa podobnie do **GET\_CARAVANS**. Różnica polega na tym, że zestaw danych ma postać **pid cid id1 d id2 l m s p**. Wartości **pid cid id1 d id2 l m s** mają to samo znaczenie, co w **GET\_CARAVANS**, natomiast **p** to ilość gotówki posiadanej przez karawanę.

Przykład:

```
> MY_CARAVANS
< OK
< 2
< 7383 435 1 0 1 5 6 7 200
< 7383 9856 5 8 4 2 2 11 142
```

## GET\_CARAVAN\_LOAD

Przyjmuje jedną liczbę – identyfikator karawany. Zwraca liczbę **t** oznaczającą liczbę różnych miast, z których pochodzą antyki wiezione przez karawanę. Potem następuje **t** zestawów danych **m n** – odpowiednio identyfikator miasta, z którego pochodzą antyki, i ich liczba. Przykład:

```
> GET_CARAVAN_LOAD 435
< OK
< 2
< 1 3
< 0 7
```

## MOVE

Przyjmuje dwie liczby – identyfikator karawany i identyfikator miasta docelowego. Miasto docelowe musi być połączone bezpośrednio z miastem, w którym obecnie znajduje się karawana. Przykład:

```
> MOVE 432 1
< OK
```

## BUY

Przyjmuje dwie liczby – identyfikator karawany i liczbę antyków, jaką karawana ma kupić w mieście, w którym aktualnie się znajduje. Przykład:

```
> BUY 200 4
< OK
```

## SELL

Przyjmuje trzy liczby – identyfikator karawany, liczbę antyków do sprzedania i identyfikator miasta, z którego pochodzą sprzedawane antyki. Przykład:

```
> SELL 200 4 0
< OK
```

## MY\_TREASURY

Zwraca liczbę będącą ilością pieniędzy na koncie handlarza. Przykład:

```
> MY_TREASURY
< OK
< 700
```

## DEPOSIT

Przyjmuje dwie liczby – identyfikator karawany stojącej w mieście z bankiem i kwotę, jaką ta karawana chce wpłacić na konto handlarza. Przykład:

```
> DEPOSIT 3 200
< OK
```

## WITHDRAW

Działa analogicznie do **DEPOSIT**, ale druga z przyjmowanych liczb to kwota, którą karawana chce wypłacić z konta handlarza. Przykład:

```
> WITHDRAW 3 200
< OK
```

## BRIBE

Przyjmuje dwie liczby – identyfikator miasta, które gracz chce przekupić, i kwotę, jaka zostanie przekazana burmistrzowi. Przykład:

```
> BRIBE 2 1000
< OK
```

## GET\_BRIBES

Przyjmuje jedną liczbę – identyfikator miasta. Zwraca liczbę **G** oznaczającą liczbę graczy, którzy wpłacili łapówkę burmistrzowi miasta o podanym identyfikatorze. Potem następuje **G** zestawów danych **u b** – odpowiednio identyfikator przekupującego gracza i wpłacona przez niego kwota łapówki. Przykład:

```
> GET_BRIBES 2
< OK
< 3
< 5647 523
< 5733 4553
< 5923 98
```

## GET\_CURFEWS

Zwraca liczbę **b** miast, w których burmistrzowie nałożyli ograniczenia na karawany. Potem następuje **b** zestawów danych **t u l** – odpowiednio identyfikator miasta, identyfikator zwycięzcy licytacji i siła blokady (1 – nie pozwala wyjeżdżać obcym karawanom z miasta, 2 – rekwiruje pieniądze i towary przy wjeździe obcych karawan do miasta). Przykład:

```
GET_CURFEWS
< OK
< 2
< 0 1 1
< 2 3 2
```

## Możliwe błędy

- 200 not your caravan – gracz próbuje wykonać operację nie swoją karawaną
- 201 not enough money in bank – gracz próbuje użyć większej ilości pieniędzy niż faktycznie ma na koncie
- 202 no such town – miasto o podanym identyfikatorze nie istnieje
- 300 caravan limit reached – gracz osiągnął limit karawan i próbuje dodać nową
- 301 no such caravan – karawana o podanym identyfikatorze nie istnieje
- 400 incorrect parameters – podane parametry karawany są niepoprawne
- 401 already moved – karawana już się ruszyła w tej turze
- 402 no such road – nie ma bezpośredniej krawędzi do danego miasta
- 403 not in town – karawana nie jest w mieście
- 404 not enough money in caravan – karawana ma za mało pieniędzy, żeby wykonać operację
- 405 not enough capacity – karawana nie ma wystarczająco wiele wolnego miejsca
- 406 not enough items – karawana ma za mało przedmiotów danego rodzaju
- 407 no such items – karawana nie ma przedmiotów danego rodzaju
- 408 no bank in town – karawana nie może wpłacić lub wypłacić
- 587 town curfewed – nie można wyjechać z miasta, bo jest zablokowane
- 600 bank in town – w mieście jest bank i nie można przekupić burmistrza