

Kobky

Robert programuje novou počítačovou hru. Ve hře vystupuje jeden hlavní hrdina, n jeho nepřátel a je tam $n + 1$ kobek. Nepřátelé jsou očíslováni od 0 do $n - 1$, kobky jsou označeny čísly od 0 do n . Nepřítel i ($0 \leq i \leq n - 1$) se nachází v kobce i a má sílu $s[i]$. Kobka n je prázdná.

Hrdina začíná hru tím, že vstoupí do kobky x . Na začátku hry má sílu z . Kdykoliv hrdina vstoupí do některé kobky i ($0 \leq i \leq n - 1$), utká se s nepřítelem i , což vede k jednomu z dvou možných výsledků:

- Má-li hrdina větší nebo stejnou sílu než jeho nepřítel (ten má sílu $s[i]$), hrdina zvítězí. Síla hrdiny se tím **zvýší** o $s[i]$ ($s[i] \geq 1$). V tomto případě hrdina půjde příště do kobky $w[i]$ ($w[i] > i$).
- V opačném případě je hrdina poražen. Síla hrdiny se tím **zvýší** o $p[i]$ ($p[i] \geq 1$). V tomto případě hrdina půjde příště do kobky $l[i]$.

Poznamenejme, že $p[i]$ může být menší než, stejné, nebo větší než $s[i]$. Také $l[i]$ může být menší než, stejné, nebo větší než i . Bez ohledu na výsledek souboje nepřítel zůstává v kobce i a zachovává si svoji sílu $s[i]$.

Hra končí, když hrdina vstoupí do kobky n . Lze ukázat, že hra skončí po konečném počtu soubojů bez ohledu na to, ve které kobce a s jakou silou hrdina hru zahájí.

Robert vás požádal, abyste jeho hru otestovali provedením q simulací. Pro každou simulaci Robert určí počáteční kobku x a počáteční sílu hrdiny z . Vaším úkolem je určit pro každou simulaci, jakou sílu bude mít hrdina na konci hry.

Implementační detaily

Implementujte následující funkce:

```
void init(int n, int[] s, int[] p, int[] w, int[] l)
```

- n : počet nepřátel.
- s , p , w , l : pole délky n . Pro $0 \leq i \leq n - 1$:
 - $s[i]$ je síla nepřítele i . Je to zároveň síla získaná hrdinou po vítězství nad nepřítelem i .
 - $p[i]$ je síla získaná hrdinou po porážce od nepřítele i .
 - $w[i]$ je číslo kobky, kam půjde hrdina po vítězství nad nepřítelem i .
 - $l[i]$ je číslo kobky, kam půjde hrdina po porážce od nepřítele i .
- Tato funkce je volána právě jednou, před všemi voláními funkce `simulate` (viz níže).

```
int64 simulate(int x, int z)
```

- x : číslo kobky, kam hrdina vstupuje na začátku hry.
- z : hrdinova počáteční síla.
- Tato funkce vrátí sílu hrdiny po skončení hry, pokud hrdina zahájil hru vstupem do kobky x s počáteční silou z .
- Funkce bude zavolána přesně q krát.

Příklad

Uvažujme následující volání:

```
init(3, [2, 6, 9], [3, 1, 2], [2, 2, 3], [1, 0, 1])
```

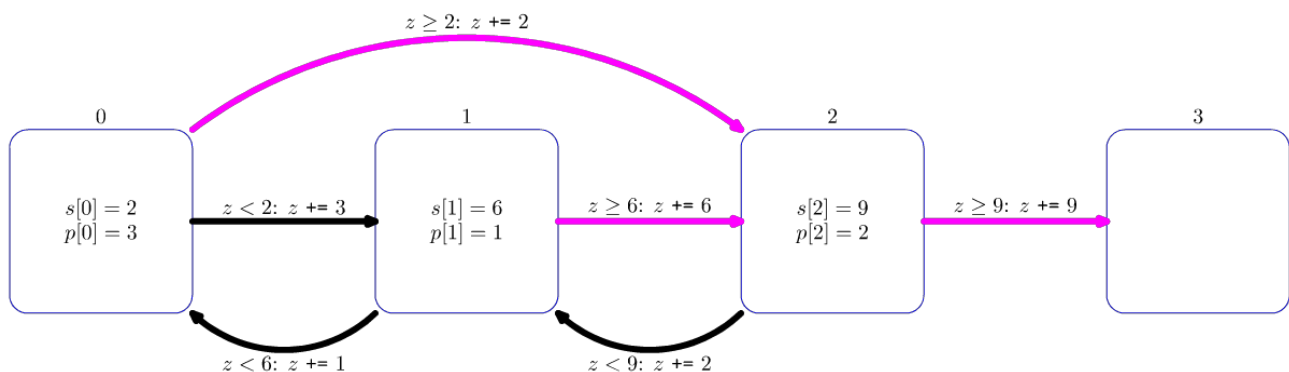


Diagram znázorňuje situaci po tomto volání. Každý čtvereček představuje jednu kobku. Pro kobky 0, 1 a 2 jsou uvnitř čtverečků uvedeny jejich hodnoty $s[i]$ a $p[i]$. Fialové šipky ukazují, kam půjde hrdina po vítězném souboji, zatímco černé šipky ukazují, kam půjde v případě porážky.

Nechť vyhodnocovač zavolá `simulate(0, 1)`.

Hra bude probíhat takto:

Kobka	Síla hrdiny před soubojem	Výsledek
0	1	Porážka
1	4	Porážka
0	5	Vítězství
2	7	Porážka
1	9	Vítězství
2	15	Vítězství
3	24	Konec hry

Funkce vrátí výslednou hodnotu 24.

Nechť vyhodnocovač zavolá `simulate(2, 3)`.

Hra bude probíhat takto:

Kobka	Síla hrdiny před soubojem	Výsledek
2	3	Porážka
1	5	Porážka
0	6	Vítězství
2	8	Porážka
1	10	Vítězství
2	16	Vítězství
3	25	Konec hry

Funkce vrátí výslednou hodnotu 25.

Omezení

- $1 \leq n \leq 400\,000$
- $1 \leq q \leq 50\,000$
- $1 \leq s[i], p[i] \leq 10^7$ (pro všechna $0 \leq i \leq n - 1$)
- $0 \leq l[i], w[i] \leq n$ (pro všechna $0 \leq i \leq n - 1$)
- $w[i] > i$ (pro všechna $0 \leq i \leq n - 1$)
- $0 \leq x \leq n - 1$
- $1 \leq z \leq 10^7$

Podúlohy

1. (11 bodů) $n \leq 50\,000$, $q \leq 100$, $s[i], p[i] \leq 10\,000$ (pro všechna $0 \leq i \leq n - 1$)
2. (26 bodů) $s[i] = p[i]$ (pro všechna $0 \leq i \leq n - 1$)
3. (13 bodů) $n \leq 50\,000$, všichni nepřátelé mají stejnou sílu, neboli $s[i] = s[j]$ pro všechna $0 \leq i, j \leq n - 1$.
4. (12 bodů) $n \leq 50\,000$, mezi hodnotami $s[i]$ se nachází nejvýše 5 různých hodnot.
5. (27 bodů) $n \leq 50\,000$
6. (11 bodů) Žádná další omezení.

Ukázkový vyhodnocovač

Ukázkový vyhodnocovač čte vstup v následujícím tvaru:

- řádek 1: $n \ q$
- řádek 2: $s[0] \ s[1] \ \dots \ s[n - 1]$
- řádek 3: $p[0] \ p[1] \ \dots \ p[n - 1]$

- řádek 4: $w[0] \ w[1] \ \dots \ w[n-1]$
- řádek 5: $l[0] \ l[1] \ \dots \ l[n-1]$
- řádek $6 + i$ ($0 \leq i \leq q-1$): $x \ z$ pro i -té volání funkce `simulate`.

Výsledek vypisuje ve tvaru:

- řádek $1 + i$ ($0 \leq i \leq q-1$): výsledná hodnota i -tého volání funkce `simulate`.