




Mali Tetris

Kot verjetno že veste je popularno raunalniško igrico "Tetris" izumil ruski programer Alexey Pajitnov. Vaša naloga je, da napišete program, ki bo sam igral poenostavljeno različico te igre.

Igrana površina je pravokotna šahovnica. Naključni liki, sestavljeni iz enotskih kvadratov, se pojavljajo na vrhu šahovnice. Igralec nato izbere vodoravno pozicijo in rotacijo lika. Nato lik pade dol po šahovnici. Smisel naloge je, da se zapolni vodoravne vrstice. Ko je cela vrstica zapolnjena, izgine in vse kocke, ki so nad njo padejo.

Pri naši izvedbi imamo šahovnico velikosti 3×4 enot in like le sledečih oblik:

Tip	Lik
1	
2	
3	

Izgubite, če na neki točki obstaja pet nepraznih vrstic šahovnice. Zmagate pa, če ne izgubite po tem, ko pade n likov.

Napisati morate program, ki igra in ne izgubi neglede na naslednji lik, ki se pojavi.

Podatki o implementaciji

Implementirati morate štiri funkcije (metode):

- `void init(int n)`. Funkcija, ki se izvede pred vsemi.
- `void new_figure(int figure_type)`. Ta funkcija je poklicana, ko se pojavi nov lik. `figure_type` je število med `1` in `3`, ki predstavlja tip lika iz zgornje tabele.
- `int getPosition()`. Ta funkcija mora vrniti število med `0` in `2`, pozicijo najbolj levega konca zadnjega lika.
- `int getRotation()`. Ta funkcija mora vrniti število med `0` in `3`, ki predstavlja število obratov lika za 90° v smeri urinega kazalca.

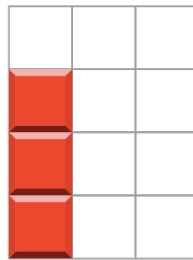
Funkciji `getPosition` in `getRotation` bosta klicani po klicu funkcije `new_figure`.

Prosimo da uporabite že narejene predložene datoteke za izbran programski jezik.

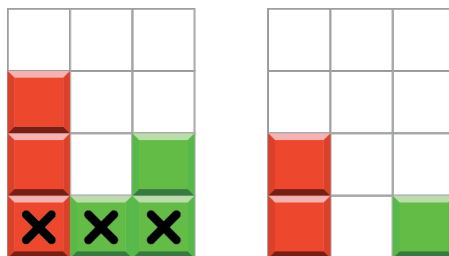
Primer

Ocenjevalnik naredi sledeče klice funkcij:

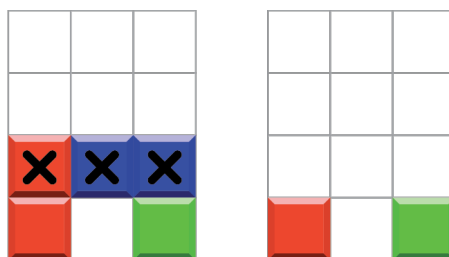
- `init(3)`. Pojavili se bodo trije liki.
- `new_figure(1)`. Lik tipa `1` pade iz vrha šahovnice.
- `getPosition()` vrne `0`. To pomeni, da želi igralec spustiti lik v skrajno lev stolpec.
- `getRotation()` vrne `1` (ali `3`). To pomeni, da želi igralec obniti lik navpično.
- Po tem vse skupaj izgleda tako:



- `new_figure(2)`.
- `getPosition()` vrne `1`.
- `getRotation()` vrne `1`.
- Po tem prva vrstica izgine, ker je prva vrstica popolnoma zapolnjena in vse skupaj izgleda takole:



- `new_figure(1)`.
- `getPosition()` vrne `1`.
- `getRotation()` vrne `0` (ali `2`).
- Po padcu tega lika izgine druga vrstica in vse skupaj izgleda takole:



Podnaloge

Pri vseh podnalogah velja $n \leq 1000$.

1. (7 točk) Vsi liki so tipa **1**,
2. (13 točk) Vsi liki so tipa **2**,
3. (21 točk) Vsi liki so tipa **3**,
4. (53 točk) Liki so različnih tipov

Ocenjevalnik

Ocenjevalnik prebere vhod v sledečem zapisu:

- vrstica 1: Eno celo število n .
- vrstica 2: n celo število: tip likov.

Opombe k jeziku

Prosimo, uporabite že narejene datoteke za več informacij k izbranem programskem jeziku.