

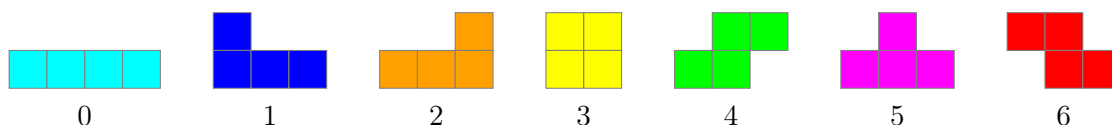
Programiranje 1 — tekmovalna naloga

Rok za oddajo: nedelja, 9. januarja 2022

Tetris in minimizacija maksimuma

Tetris

Kot nemara veste, je Tetris igra, pri kateri v pravokotno jamo spuščamo like, prikazane na sliki 1. V vsakem ciklu se na sredini na vrhu jame naključno prikaže eden od likov, igralec pa ga lahko med enakomernim spuščanjem premika levo in desno ter vrti za 90 stopinj v smeri urinega kazalca. Ko se lik spusti na dno jame ali na že zgrajeno strukturo, se odstranijo morebitne vrstice, ki so se v celoti zapolnile, nato pa se prične naslednji cikel. Pri odstranjevanju vrstic se lahko zgodi, da kak del strukture obvisi v zraku. To je v nasprotju s fizikalnimi zakonitostmi, vendar pa se pri Tetrisu glede takih »malenkosti« ne vznemirjamo.



Slika 1: Tetrisovi liki.

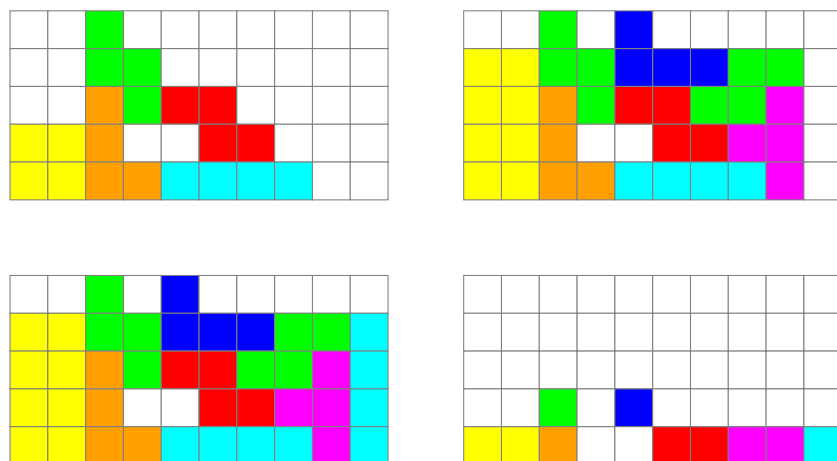
Standardna Tetrisova jama je široka 10, visoka pa 20 enot. V tej nalogi bo širina jame prav tako enaka 10, njena višina pa ne bo omejena.

Naj trojka (ℓ, o, x) predstavlja spust lika z indeksom ℓ v orientaciji o na koordinato x . Indeksi likov so prikazani na sliki 1, vrednost o je lahko enaka 0 (izhodiščna orientacija), 1 (rotacija lika v izhodiščni orientaciji za 90 stopinj v smeri urinega kazalca), 2 (rotacija za 180 stopinj) ali 3 (rotacija za 270 stopinj), koordinate x pa segajo od 0 (skrajno levi stolpec) do 9. Če lik širine w spustimo na koordinato x , se bo njegov skrajno levi stolpec ob pristanku nahajal na koordinati x , skrajno desni pa na koordinati $x + w - 1$.

Oglejmo si primer. Denimo, da pričnemo s prazno Tetrisovo jamo in po vrsti izvršimo spuste $(2, 1, 2)$, $(4, 1, 2)$, $(0, 0, 4)$, $(6, 2, 4)$, $(3, 2, 0)$, $(5, 3, 7)$, $(4, 0, 6)$, $(1, 0, 4)$, $(3, 1, 0)$ in $(0, 3, 9)$. Slika 2 prikazuje vsebino jame po prvih petih spustih (zgoraj levo), po prvih devetih spustih (zgoraj desno) in po vseh desetih spustih (spodaj). Ko v jamo spustimo deseti lik, se zapolnijo tri vrstice — druga, tretja in peta (slika 2, spodaj levo). Vse tri vrstice takoj izbrišemo (slika 2, spodaj desno).

Naloga

Napišite program, ki za podano zaporedje indeksov likov določi orientacije in koordinate spustov posameznih likov tako, da bo višina najvišje ležeče zapolnjene celice, ki se bo pojavila po kateremkoli zaključenem spustu, najmanjša možna. Mero uspešnosti vašega programa bomo torej izračunali tako, da bomo (1) vhodne like po vrsti spustili v jamo, kot bo narekoval izhod vašega programa; (2) po vsakem spustu ugotovili višino najvišje ležeče zapolnjene celice; (3) poiskali najmanjšo izmed dobljenih višin. Višino bomo vsakokrat izmerili po brisanju morebitnih zapolnjenih vrstic. Na primer, po izvedbi vseh desetih spustov v primeru s slike 2 je višina najvišje ležeče celice enaka 2, ne 5. Končni rezultat za ta primer pa je kljub temu enak 5, saj je vsebina jame vsaj enkrat dosegla to višino (npr.



Slika 2: Primer zaporedja spustov.

po drugem spustu).

Vhod

V prvi vrstici je podano število spustov ($n \in [1, 1000]$), v vsaki od naslednjih n vrstic pa je podan indeks lika (število z intervala $[0, 6]$).

Izhod

Izpišite n vrstic, pri čemer naj i -ta vrstica (za $i \in \{1, \dots, n\}$) vsebuje dve števili, ločeni s presledkom:

- orientacijo $o \in [0, 3]$, v kateri naj se spusti i -ti lik;
- koordinato $x \in [0, 9]$, na katero naj pade i -ti lik.

Javni testni primer

Vhod:

```
10
2
4
0
6
3
5
4
1
3
0
```

Izhod (eden od številnih možnih):

```
1 2
1 2
0 4
2 4
2 0
```

```
3 7
0 6
0 4
1 0
3 9
```

Ocenjevanje

Vsak oddani program bomo preizkusili z isto množico testnih primerov. Za vsak testni primer bomo izračunali mero uspešnosti vsakega posameznega programa. Vašemu programu bomo za vsak testni primer dodelili $N - k$ točk, kjer je N število prispelih programov, k pa število programov, ki je na tem testnem primeru doseglo strogo boljši rezultat od vašega. Če bo vaš program proizvedel neveljaven izhod (izhod v napačni obliki ali pa neveljavno orientacijo ali koordinato za nek lik), bo za pripadajoči testni primer prejel 0 točk. Programe bomo na koncu razvrstili preprosto po padajočem številu točk.

Dovoljena in nedovoljena sredstva

Kršitev sledečih pravil pomeni diskvalifikacijo, kršitev prvega pa še kaj več:

- **Sleherna vrstica programa mora biti vaša in samo vaša.** Lahko si pomagata s spletnimi viri, nikakor pa ne smete od tam (ali od koderkoli drugod) prepisovati oz. kopirati programske kode. Kršitev tega pravila se bo štela kot goljufanje, to pa ne bo pomenilo samo diskvalifikacije, ampak tudi obravnavo pred fakultetno disciplinsko komisijo (z verjetno enoletno prepovedjo opravljanja izpita).
- Uporabljate lahko samo pakete `java.lang` in `java.util` (brez teh ne boste mogli shajati) ter `java.util.functional` in `java.util.stream` (če ste morda navajeni sodobnih funkcijskih konstruktorov).
- Vaš program ne sme tvoriti procesov ali niti.
- Vaš program ne sme izdelovati datotek. Piše lahko izključno na standardni izhod.

Nagrade

Avtor prvouvrščenega programa bo pri predmetu Programiranje 1 neposredno prejel oceno 10 in bo s tem oproščen opravljanja izpita (tako pisnega kot ustnega). Avtorji, ki bodo zavzeli mesta od drugega do desetega, pa bodo na izpitu prejeli dodatne točke:

Mesto	Dodatne točke
2.	25
3.	20
4.	15
5.	10
6.–10.	5

Na izpitu bo možno zbrati 100 točk, zato se bodo nagrade lahko kar pošteno občutile.

Oddaja programa

Program oddajte v obliki ene same datoteke z nazivom `DNTE_vvvvvvvvv.java`, pri čemer niz `vvvvvvvvv` zamenjajte s svojo vpisno številko. (Morebitne dodatne razrede definirajte kot statične notranje razrede v razredu `DNTE_vvvvvvvvv`.)