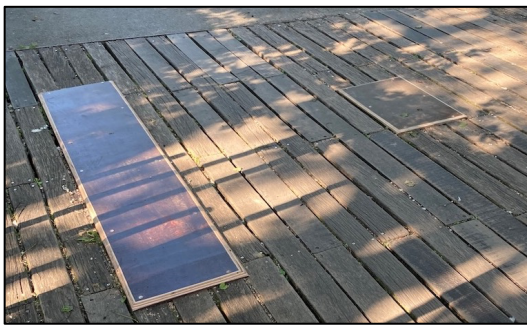
Izpit iz predmeta Programiranje 1, 11. junij 2024 **Čas pisanja: ~~60 minut~~ → 90 minut**

Dovoljena je uporaba literature na poljubnih medijih, prepovedana pa je komunikacija in uporaba inteligentnih pripomočkov.

Ob koseškem bajerju gre kolesarska steza čez dotrajan most, ki ga Oddelek za motorni promet in gospodarske dejavnosti MOL krpa z lesenimi ploščami. Seznami plošč so podani kot terke (x0, y0, x1, y1), ki predstavljajo koordinate nasprotnih oglišč (x0 < x1, y0 < y1). **Vrstni red elementov je enak vrstnemu redu, v katerem so dodajali plošče.** Plošča pokriva točke od x0, y0 do x1, y1, vendar **brez x1 in y1**. Pravokotnik, ki sega od 3 do 5, je tako dolg 2, ne 3. **Plošče se lahko prekrivajo.**

V testih je že podana funkcija **`presek(plosce, i, j)`**, ki vrne **`True`**, če se i-ta in j-ta plošča (vsaj delno) prekrivata.

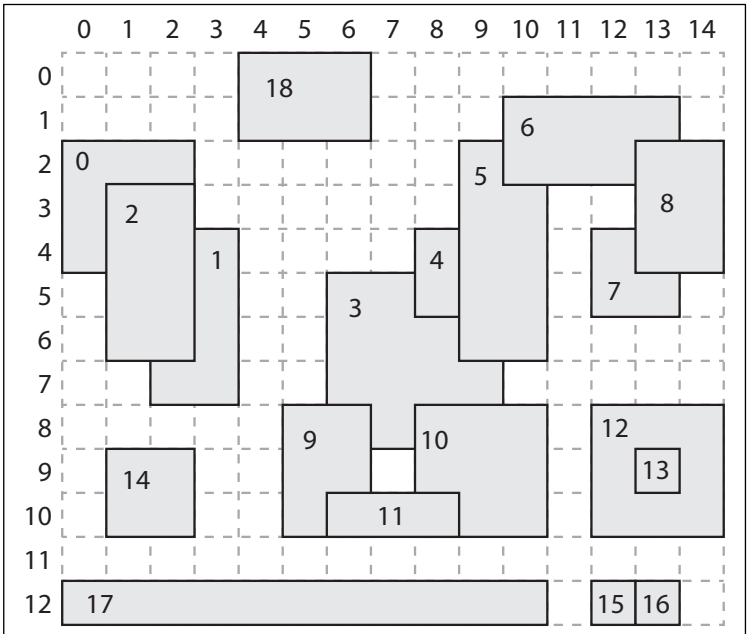


Gornja slika je lanska; zdaj je plošč več. Zaradi rež in spolzke podlage je bil most izziv že prej, po novem pa je treba voziti še slalom. Most bo obnovljen v sklopu že sedem let obljubljane "celovite prenove območja" za katero "še ni časovnice". Rešili so le problem ilegalnega parkiranja in sicer tako, da so odstranili prometni znak za prepoved parkiranja.

1. Pokritost

Napiši funkcijo `pokritost(plosce)`, ki prejme seznam plošč in vrne par (terko) ploščino pokritega dela mostu in koordinato, nad katero je največ plošč. Če je takšnih koordinat več, vrne poljubno izmed njih.

Za primer na sliki mora vrniti (93, (2, 4)), saj plošče pokrivajo 103 kvadratkov, največkrat (trikrat) pa je pokrita točka (2, 4).



Številke plošč so enake indeksom v seznamu. Plošče z višjimi številkami so bile na most dodane kasneje kot plošče z nižjimi.

2. Inventar

Ker so plošče menda *primer dobre prakse*, jim bo MOL opremil z inventarnimi številkami. Predtem je potrebno narediti popis, zato napiši funkcijo `inventar(plosce, ime_dat)`, ki v datoteko s podanim imenom shrani dimenzije in število kosov plošč posamezne oblike. Datoteka mora biti oblikovana natančno tako, kot kaže datoteka plosce.txt v testih.

Dimenzije uredi tako, da bo prva vedno manjša. Plošči dimenzij, na primer, 2x3 in 3x2 sta enaki, zato obe upoštevaj, kot da sta veliki 2x3.

3. Stiki

Napiši funkcijo `neposredno_na(plosce, i)`, ki vrne množico indeksov plošč, ki se nahajajo neposredno nad i-to.

Klic `neposredno_na(plosce, 3)` vrne `{4, 9, 10}`. Plošča 5 ni neposredno nad 3, ker je vmes 4. Klic `neposredno_na(plosce, 9)` vrne `{11}`; plošča 3 ni nad 9, ker je pod njo.

4. Nosilnost

Napiši funkcijo `nosilnost(plosce, i)`, ki vrne skupno ploščino i-te plošče in vseh plošč, ki so *neposredno ali posredno* nad njo. Plošča je posredno nad drugo, če je privita v katero od plošč, ki so (neposredno ali posredno) privite vanjo.

Za ploščo 0 mora funkcija vrniti 25 (3x3 + 2x4 + 2x4). Za ploščo 3 dovolim, da vaša funkcija deluje narobe in ploščo 11 šteje dvakrat, seveda pa bom cenil, če jo bo le enkrat. (Pravilna rešitev je 65, priznal pa bom tudi 65.)

OBRNI LIST

5. Slalom

Napiši razred `Most`:

- konstruktor prejme seznam plošč,
- metoda `pokrito(x, y)` pa vrne `True`, če je podana koordinata pokrita.

Napiši razred `Kolesar`:

- konstruktor dobi objekt razreda Most in začetne koordinate kolesarja (njegovi argumenti bodo torej `(self, most, x, y).`)
- `premik(smer)` prejme niz "`<`", "`>`", "`^`" ali "`v`" in premakne kolesarja za eno polje v to smer, če na ciljnem polju ni plošče. Če je tam plošča, kolesar pade; po padcu se ne premika več, torej kasnejši klici metode `premik` ne naredijo več ničesar. (^ pomeni, da se koordinata y zmanjša.)
- `pozicija()` vrne trenutne koordinate kolesarja.