

Izpit pri predmetu Programiranje 2 — 10. julij 2017

Čas reševanja: 90 minut.

Vse naloge so enakovredne.

Rešitev naloge i (za $i \in \{1, 2, 3\}$) oddajte v datoteki `vvvvvvvvv_i.c`, kjer je `vvvvvvvvv` vaša vpisna številka. Na primer: `63160999_1.c`, `63160999_2.c`, `63160999_3.c`.

- ① Napišite program, ki prebere števili n in k in izpiše zmnožek števila n s številom k .

Vhod:

Na vhodu sta zapisani celi števili $n \in [1, 10^{1000}]$ in $k \in [2, 9]$, ločeni s presledkom.

V testnih primerih J1–J5 in S1–S25 za vsako števko n_i v številu n velja $n_i k < 10$.

Izhod:

Izpišite zmnožek.

Javni primer 1 (vhod/izhod):

12303120103002 3

36909360309006

- ② Robot se premika po koordinatni mreži velikosti 8×8 . Prične v *spodnji desni* (!) točki mreže, nato pa sledi ukazom 0 (premik levo), 1 (premik navzgor), 2 (premik desno) in 3 (premik navzdol). Če je ukaz v trenutnem položaju mogoče izvesti, se v zahtevani smeri premakne za eno enoto, sicer pa ga ignorira.

Recimo, da točke, ki jih je robot na svoji poti vsaj enkrat obiskal (vključno z izhodiščno točko) označimo z 1, ostale točke pa z 0. Če oznake preberemo po vrsticah od zgoraj navzdol, vsako vrstico pa od leve proti desni, dobimo 64-bitno število. Vaš program naj prebere število n in zaporedje n ukazov, izpiše pa naj pretvorbo dobljenega 64-bitnega števila v desetiški sistem.

Vhod:

V prvi vrstici je zapisano celo število $n \in [1, 1000]$, v drugi pa je nanizanih n števil iz množice $\{0, 1, 2, 3\}$, ki predstavljajo posamezne ukaze. Ta števila so med seboj ločena s presledkom.

V primerih J1–J5 in S1–S25 ni neveljavnih ukazov (vse ukaze je v danem položaju mogoče izvesti).

Izhod:

Izpišite zahtevano število.

Primer je prikazan na naslednji strani.

Javni primer 6 (vhod/izhod):

```
9
0 1 0 0 3 3 1 2 1
```

```
265739
```

V tem primeru se šesti ukaz ignorira. Matrika obiskanosti na koncu robotovega potovanja izgleda tako:

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0
0 0 0 0 1 1 1 0
0 0 0 0 1 0 1 1
```

- ③ Napišite program, ki prebere število n in n nizov, sestavljenih iz znakov {, [, (,),] in }, ter izpiše število nizov, ki predstavljajo pravilno gnezden oklepajni izraz. Tovrstni izrazi so definirani s sledečimi pravili:

- Prazen niz je pravilno gnezden oklepajni izraz.
- Če je niz S pravilno gnezden oklepajni izraz, so takšni tudi nizi (S) , $[S]$ in $\{S\}$.
- Če sta niza S in T pravilno gnezdena oklepajna izraza, potem to velja tudi za niz ST .
- Ne obstaja noben drug pravilno gnezden oklepajni izraz.

Vhod:

V prvi vrstici je zapisano celo število $n \in [1, 1000]$, nato pa sledi n vrstic z nizi dolžine od 1 do vključno 1000. Nizi vsebujejo samo znake {, [, (,),] in }.

V testnih primerih J1–J5 in S1–S25 so nizi sestavljeni samo iz znakov (in).

Izhod:

Izpišite število pravilno gnezdenih oklepajnih izrazov.

Javni primer 6 (vhod/izhod):

```
5
[]
([])
({[[{}}]])[()]
[{}]{(){} }
())[()]( )
```

```
3
```

V gornjem primeru so pravilno gnezdeni prvi, tretji in četrti izraz.