



Robotivõistlus

Szegedi ülikooli teadlased korraldavad robotite programmeerimise võistlust, millel osaleb ka sinu sõber Hanga. Ülesanne on Ungari targa karjakoeratõu puli auks programmeerida parim *Pulibot*.

Puliboti testitakse $(H + 2) \times (W + 2)$ ruudust koosnevas labürindis. Ruudustiku read on nummerdatud -1 kuni H põhjast lõunasse ja tulpad -1 kuni W läänest itta. Viitame reas r ja tulpas c olevale ruudule (kus $-1 \leq r \leq H$ ja $-1 \leq c \leq W$) kui (r, c) .

Vaatame ruutu (r, c) , kus $0 \leq r < H$ ja $0 \leq c < W$. Selle ruudu **kõrval** on 4 ruutu:

- ruut $(r, c - 1)$ on ruudust (r, c) **läänes**;
- ruut $(r + 1, c)$ on ruudust (r, c) **lõunas**;
- ruut $(r, c + 1)$ on ruudust (r, c) **idas**;
- ruut $(r - 1, c)$ on ruudust (r, c) **põhjas**.

Ruutu (r, c) kutsutakse **servaruuduks**, kui kehtib $r = -1$ või $r = H$ või $c = -1$ või $c = W$. Iga ruut, mis ei ole servaruut, kas sisaldab **takistust** või on **tühj** ruut. Lisaks on igal tühjal ruudul **värv**: mittenegatiivne täisarv lõigust 0 kuni Z_{MAX} (kaasaarvatud). Alguses on kõigi tühjade ruutude värv 0.

Vaatame näiteks labürinti, kus $H = 4$ ja $W = 5$ ning kus on üks takistus ruudus $(1, 3)$:

| | -1 | 0 | 1 | 2 | 3 | 4 | 5 |
|----|----|---|---|---|---|---|---|
| -1 | | | | | | | |
| 0 | | 0 | 0 | 0 | 0 | 0 | |
| 1 | | 0 | 0 | 0 | | 0 | |
| 2 | | 0 | 0 | 0 | 0 | 0 | |
| 3 | | 0 | 0 | 0 | 0 | 0 | |
| 4 | | | | | | | |

Ainus takistusega ruut on tähistatud ristiga. Servaruudud on joonisel tumedad. Arv tühjas ruudus tähistab ruudu värvi.

Teekond likkusega ℓ (kus $\ell > 0$) ruudust (r_0, c_0) ruutu (r_ℓ, c_ℓ) on paarikaupa erinevate **tühjade** ruutude jada $(r_0, c_0), (r_1, c_1), \dots, (r_\ell, c_\ell)$, kus iga $0 \leq i < \ell$ korral on ruudud (r_i, c_i) ja (r_{i+1}, c_{i+1}) kõrvuti.

Pane tähele, et teekonnas pikkusega ℓ on täpselt $\ell + 1$ ruutu.

Võistlusel on labürint selline, et ruudust $(0, 0)$ ruutu $(H - 1, W - 1)$ leidub vähemalt üks teekond. See tähendab ka, et ruudud $(0, 0)$ ja $(H - 1, W - 1)$ on kindlasti tühjad.

Hanga ei tea, missugused ruudud on tühjad ja missugustes on takistused.

Sinu ülesanne on aidata Hangal programmeerida Pulibot nii, et see suudab leida *lühima teekonna* (lühima pikkusega teekonna) ruudust $(0, 0)$ ruutu $(H - 1, W - 1)$ võistlusel üles seatud tundmatus labüridis. Järgnevad Puliboti spetsifikatsioon ja võistluse reeglid.

Pane tähele, et selle ülesande teksti viimane osa kirjeldab visualiseerimistööriista, mida saad Puliboti visualiseerimiseks kasutada.

Puliboti spetsifikatsioon

Olgu ruudu (r, c) (kus $-1 \leq r \leq H$ ja $-1 \leq c \leq W$) **olek** täisarv:

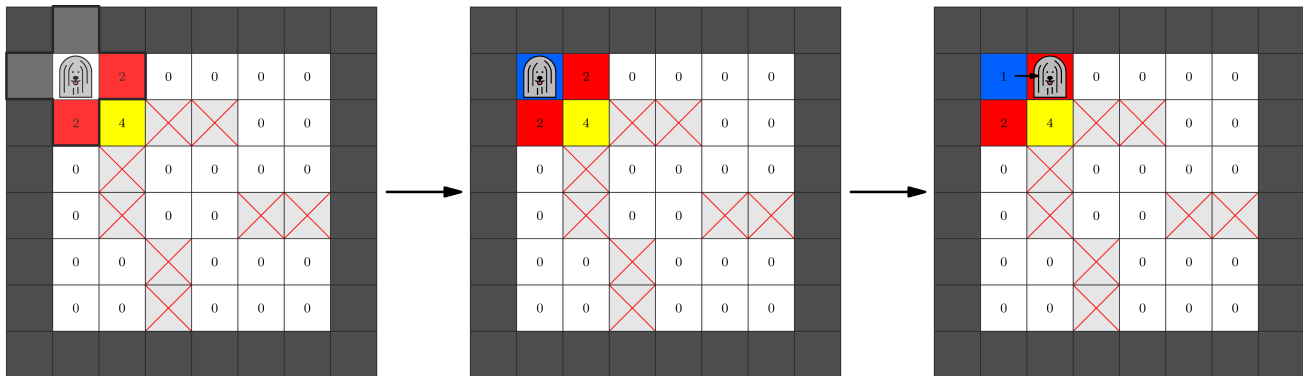
- kui ruut (r, c) on servaruut, siis on selle olek -2 ;
- kui ruudus (r, c) on takistus, siis on selle olek -1 ;
- kui ruut (r, c) on tühi ruut, siis on selle olek ruudu värv.

Puliboti programmi täidetakse sammude jadana. Igal sammul saab Pulibot teada lähedalolevate ruutude olekud ja täidab seejärel käsu, mis sõltub ruutude olekutest. Täpsemalt teeb ta järgmist.

Olgu antud sammu alguses Pulibot tühjal ruudul (r, c) . Programmi samm tehakse järgmiselt:

1. Esiteks tuvastab Pulibot **olekumassiivi**, s.t massiivi $S = [S[0], S[1], S[2], S[3], S[4]]$, mis koosneb ruudu (r, c) ja kõigi selle kõrval olevate ruutude olekutest:
 - $S[0]$ on ruudu (r, c) olek;
 - $S[1]$ on läänes asuva ruudu olek;
 - $S[2]$ on lõunas oleva ruudu olek;
 - $S[3]$ on idas oleva ruudu olek;
 - $S[4]$ on põhjas oleva ruudu olek.
2. Seejärel teeb Pulibot kindlaks **juhise** (Z, A) , mis vastab tuvastatud olekumassiivile.
3. Lõpuks täidab Pulibot selle juhise: määrab ruudu (r, c) värviks Z ja täidab käsu A , mis on üks järgmistest:
 - *seisa paigal* ruudus (r, c) ;
 - *liigu* ühte 4 kõrvalolevast ruudust;
 - *lõpeta programmi töö*.

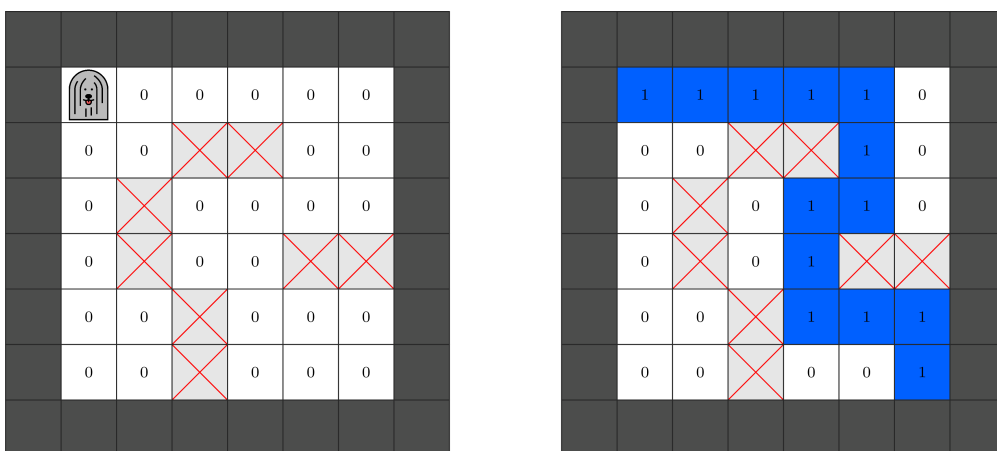
Vaatame näiteks järgmisel joonisel vasakpoolses osas toodud juhtu. Pulibot on praegu ruudus $(0, 0)$ värviga 0. Pulibot tuvastab olekumassiivi $S = [0, -2, 2, 2, -2]$. Pulibotil võib olla programm, mis selle massiivi peale määrab praeguse ruudu värviks $Z = 1$ ja liigub seejärel itta, nagu näidatud keskmisel ja paremal jooniseosal:



Robotivõistluse reeglid

- Kõige alguses on Pulibot ruudus $(0, 0)$ ja hakkab programmi käivitama.
- Pulibot ei tohi liikuda ruutu, mis ei ole tühi.
- Puliboti programm peab töö lõpetama ülimalt 500 000 sammu järel.
- Pärast programmi töö lõppu peavad tühjad ruudud labürintis olema värvitud järgmiselt:
 - Leidub lühim teekond ruudust $(0, 0)$ ruutu $(H - 1, W - 1)$, kus teekonna iga ruudu värv on 1.
 - Kõigi teiste tühjade ruutude värv on 0.
- Pulibot võib programmi töö lõpetada mistahes tühjas ruudus olles.

Näiteks järgmine joonis näitab võimalikku labürinti, kus $H = W = 6$. Algseisu kujutatakse vasakul ja tühjade ruutude üht võimalikku värvimist pärast programmi töö lõppu paremal:



Realisatsioon

Lahendusena tuleb realiseerida funktsioon

```
void program_pulibot()
```

- See funktsioon peaks looma Puliboti programmi. Programm peab töötama korrektselt kõigi H ja W väärtuste ja iga ülesande tingimustele vastava labürinti jaoks.
- Seda funktsiooni käivitatakse igas testis täpselt üks kord.

Funktsioon võib välja kutsuda järgmist funktsiooni, et luua Puliboti programm:

```
void set_instruction(int[] S, int Z, char A)
```

- S : massiiv pikkusega 5, kirjeldab olekumassiivi.
- Z : mittenegatiivne täisarv, tähistab värvi.
- A : üks tähemärk, tähistab käsku Pulibotile järgmiselt:
 - H: seisa paigal;
 - W: liigu läände;
 - S: liigu lõunasse;
 - E: liigu itta;
 - N: liigu põhja;
 - T: lõpeta programmi töö.
- Selle funktsiooni väljakutsumine juhendab Puliboti, et olekumassiivi S tuvastamisel on vaja täita juhised (Z, A) .

Selle funktsiooni mitu korda sama olekumassiivi S väärtusega välja kutsumine põhjustab hindaja väljundi `Output isn't correct`.

Funktsiooni `set_instruction` ei pea välja kutsuma iga võimaliku olekumassiiviga S . Kui aga Pulibot hiljem tuvastab olekumassiivi, mille jaoks ei ole seatud juhiseid, on hindaja väljundiks `Output isn't correct`.

Pärast `program_pulibot` töö lõppu käivitatakse Puliboti programm ühes või enamas labürindis. Nende käivituste tööaeg *ei lähe arvesse* sinu lahenduse tööajana. Hindaja *ei ole adaptiivne*, s.t igas testis kasutatavad labürinid on eelnevalt määratud.

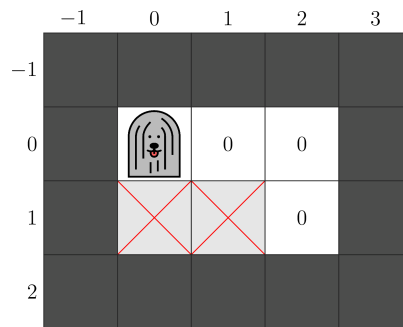
Kui Pulibot rikub enne töö lõpetamist mõnda robotivõistluse reeglit, siis on hindaja väljund `Output isn't correct`.

Näide

Funktsioon `program_pulibot` võib kutsuda välja funktsiooni `set_instruction` järgnevalt:

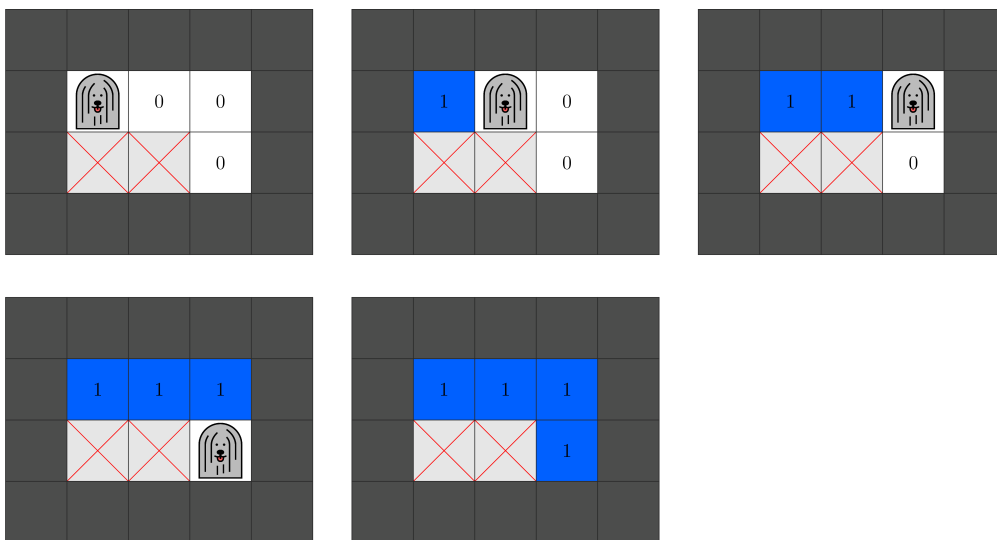
| Väljakutse | Juhis olekumassiivi S jaoks |
|--|---|
| <code>set_instruction([0, -2, -1, 0, -2], 1, E)</code> | Määra värviks 1 ja liigu itta |
| <code>set_instruction([0, 1, -1, 0, -2], 1, E)</code> | Määra värviks 1 ja liigu itta |
| <code>set_instruction([0, 1, 0, -2, -2], 1, S)</code> | Määra värviks 1 ja liigu lõunasse |
| <code>set_instruction([0, -1, -2, -2, 1], 1, T)</code> | Määra värviks 1 ja lõpeta programmi töö |

Vaatame olukorda, kus $H = 2$ ja $W = 3$ ning labürint on kujutatud järgmisel joonisel:



Selles labürendis töötab Puliboti programm neli sammu. Puliboti tuvastatavad olekumassiivid ja täidetud juhised vastavad täpselt neljale `set_instruction` väljakutsele ülalkirjeldatud järjekorras. Viimane neist lõpetab programmi töö.

Allolev joonis näitab labürinti enne igat sammu ja pärast töö lõppu.



Pane aga tähele, et see 4 juhisega programm ei pruugi leida lühimat teekonda teistes korrektsetes labürintides. Seega oleks hindaja väljundiks `Output isn't correct`.

Piirangud

$Z_{MAX} = 19$. Seega saab Pulibot kasutada värve 0 kuni 19 (kaasaarvatud).

Iga Puliboti testimiseks kasutatava labürinti jaoks:

- $2 \leq H, W \leq 15$
- Leidub vähemalt üks teekond ruudust $(0, 0)$ ruutu $(H - 1, W - 1)$.

Alamülesanded

1. (6 punkti) Labürendis pole ühtegi takistust.
2. (10 punkti) $H = 2$
3. (18 punkti) Iga tühjade ruutude paari vahel leidub täpselt üks teekond.

4. (20 punkti) Iga lühim teekond ruudust $(0,0)$ ruutu $(H-1, W-1)$ on pikkusega $H + W - 2$.

5. (46 punkti) Lisapiirangud puuduvad.

Kui mõnes testis funktsiooni `set_instruction` väljakutsed või Puliboti programm käivitamise ajal ei vasta osas Realisatsioon välja toodud piirangutele, on vastava alamülesande skooriks 0.

Igas alamülesandes saab osalisi punkte, luues värvimise, mis on peaaegu õige.

Täpsemini:

- Testjuhu lahendus on **täielik**, kui tühjade ruutude värvid programmi töö lõppedes rahuldavad robotivõistluse reegleid.
- Testjuhu lahendus on **osaline**, kui tühjade ruutude värvid programmi töö lõppedes rahuldavad järgmisi tingimusi:
 - Leidub lühim teekond ruudust $(0,0)$ ruutu $(H-1, W-1)$, mille iga ruudu värv on 1.
 - Ühegi teise tühja ruudu värv ei ole 1.
 - Mõne tühja ruudu värv on midagi muud kui 0 ja 1.

Kui lahendus ei ole täielik ega osaline, on vastava testjuhu skoor 0.

Alamülesannetes 1-4 saab täislahendus 100% ja osaline lahendus 50% vastava alamülesande punktidest.

Alamülesandes 5 oleneb punktisumma Puliboti programmis kasutatud värvide hulgast. Täpsemini: olgu Z^* maksimaalne Z väärtus üle kõigi `set_instruction` kutsete. Testjuhu skoori arvutatakse järgmise tabeli alusel:

| Tingimus | Skoor (täielik) | Skoor (osaline) |
|-----------------------|-------------------|-------------------|
| $11 \leq Z^* \leq 19$ | $20 + (19 - Z^*)$ | $12 + (19 - Z^*)$ |
| $Z^* = 10$ | 31 | 23 |
| $Z^* = 9$ | 34 | 26 |
| $Z^* = 8$ | 38 | 29 |
| $Z^* = 7$ | 42 | 32 |
| $Z^* \leq 6$ | 46 | 36 |

Iga alamülesande skoor on antud alamülesande testide eest saadud punktide miinimum.

Hindamisprogramm

Näidishindaja loeb sisendit järgmises vormingus:

- rida 1: $H \ W$

- rida $2 + r$ ($0 \leq r < H$): $m[r][0] \ m[r][1] \ \dots \ m[r][W - 1]$

m on siin massiiv H massiivist, igaühes W täisarvu, mis kirjeldavad labürindi mitte-servaruute. $m[r][c] = 0$ kui ruut (r, c) on tühi ja $m[r][c] = 1$ kui ruudus (r, c) on takistus.

Näidishindaja kutsub kõigepealt välja `program_pulibot()`. Kui näidishindaja tuvastab protokollirikkumise, kuvab ta `Protocol Violation: <MSG>` ja lõpetab töö, kus `<MSG>` on üks järgmistest veateadetest:

- `Invalid array`: S pikkus ei ole 5 või mõne i korral ei kehti $-2 \leq S[i] \leq Z_{MAX}$.
- `Invalid color`: tingimus $0 \leq Z \leq Z_{MAX}$ ei kehti.
- `Invalid action`: tähemärk A ei ole H, W, S, E, N või T.
- `Same state array`: `set_instruction` kutsuti sama massiiviga S välja vähemalt kaks korda.

Vastasel juhul kui `program_pulibot` lõpetab töö, siis käivitab näidishindaja Puliboti programmi sisendis kirjeldatud labürindi jaoks.

Näidishindaja loob kaks väljundit.

Esiteks kirjutab näidishindaja logi Puliboti tegevusest töökataloogi `robot.bin` faili. See fail on sisendiks visualiseerimistööriistale, mida kirjeldab järgmine osa.

Teiseks: kui Puliboti programm ei lõpeta edukalt tööd, siis kuvab näidishindaja ühe järgmistest veateadetest:

- `Unexpected state`: Pulibot tuvastas olekumassiivi, millega pole funktsiooni `set_instruction` välja kutsutud.
- `Invalid move`: juhise täitmise tulemusena liikus Pulibot mittetühja ruutu.
- `Too many steps`: Pulibot tegi programmi tööd lõpetamata 500 000 sammu.

Vastasel juhul olgu $e[r][c]$ ruudu (r, c) olek pärast Puliboti programmi töö lõppu. Näidishindaja kuvab H rida järgmises vormingus:

- rida $1 + r$ ($0 \leq r < H$): $e[r][0] \ e[r][1] \ \dots \ e[r][W - 1]$

Visualiseerimistööriist

Selle ülesande abifailide pakis on fail nimega `display.py`. Käivitamisel kuvab see programm Puliboti tegevused näidishindaja sisendi labürindis. Selle jaoks peab töökataloogis olema fail `robot.bin`.

Tööriista kasutamiseks käivita järgmine käsk:

```
python3 display.py
```

Näed lihtsat graafilist kasutajaliidest. Peamised võimalused on järgmised:

- Saad vaadata kogu labüri olekut. Puliboti hetkeasukoht on esiletoodud ristkülikus.
- Saad Puliboti samme läbi käia, vajutades nooltega nuppe või nende kiirklahve. Võid ka hüpata kindla sammu juurde.
- Puliboti programmi järgmine samm on välja toodud allosas. See näitab antud hetke olekumassiivi ja täidetavat juhust. Pärast viimast sammu näitab see üht hindaja veateadetest või Terminated, kui programm lõpetab edukalt töö.
- Iga numbri jaoks, mis esindab värvi, saad määrata visuaalse taustavärvi ja ka kuvatava teksti. Kuvatav tekst on lühike sõne, mis kirjutatakse igasse seda värvi ruutu. Võid taustavärve ja kuvatekste määrata ühel järgmistest viisidest:
 - määra nad dialoogiaknas pärast nupu Colors vajutamist;
 - muuda faili colors.txt sisu.
- Faili robot.bin uuesti laadimiseks kasuta nuppu Reload. See on kasulik, kui faili robot.bin sisu on muutunud.