Tietojenkäsittelytieteen valintakoe, kevät 2022

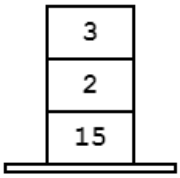
Tehtävä 3: Pinokone

Laskettaessa jonkin aritmeettisen lausekkeen arvo perinteisellä laskimella tarvittavat lukuarvot ja laskutoimitukset syötetään ensin laskimeen samassa järjestyksessä kuin ne esiintyvät itse lausekkeessa. Kun sen jälkeen painetaan "="-näppäintä, laskin laskee lausekkeen arvon. Esimerkiksi lausekkeen $15 - 2 \cdot 3$ arvo saadaan lasketuksi näppäilemällä

```
15-2*3=
```

minkä jälkeen laskin tulostaa oikean vastauksen 9.

Pinolaskin on vaihtoehtoinen tapa suorittaa tällaisia laskutoimituksia. Pinolaskimen keskeinen piirre on *pino*, johon tallennetaan tarvittavat lukuarvot ja laskennassa syntyvät tulokset. Pino voi sisältää yhden tai useamman lukuarvon, tai se voi olla tyhjä. Lukujen voi mieltää olevan pinossa samaan tapaan päällekkäin kuin esimerkiksi laatikot niitä pinottaessa. Lisättäessä uusi luku pinoon se tulee aina päällimmäiseksi, ja vastaavasti pinosta voidaan poistaa päällimmäinen luku mutta ei sen alla olevia (ennen kuin päällä oleva on ensin poistettu). Myös laskutoimitukset kohdistuvat pinon päällä oleviin lukuihin, kuten pian tarkemmin näemme. Esimerkiksi allaoleva kuva esittää pinoa, jossa on päällimmäisenä luku 3, alimpana luku 15 ja niiden välissä luku 2:



Tarkastellaan aluksi hyvin yksinkertaista pinolaskinta, jossa on seuraavat käskyt:

- `PUSH` x: laita pinon päälle luku x; tässä x on jokin kokonaisluku
- `ADD`: poista pinosta kaksi päällimmäistä lukua, laske ne yhteen ja laita summa pinon päälle
- `MUL`: poista pinosta kaksi päällimmäistä lukua, kerro ne keskenään ja laita tulo pinon päälle
- `SUB`: poista pinosta kaksi päällimmäistä lukua, vähennä päällä ollut luku alempana olleesta ja laita erotus pinon päälle.

Käskyt `ADD`, `MUL` ja `SUB` edellyttävät, että pinossa on vähintään kaksi lukua. Muuten laskin menee virhetilaan ja sen toiminta päättyy.

Yksinkertaisuuden vuoksi jätämme tästä laskimesta pois jakolaskun, jota emme jatkossa tarvitse.

Esimerkiksi lausekkeen $15 - 2 \cdot 3$ arvo saadaan lasketuksi suorittamalla käskyjono

```
PUSH 15
PUSH 2
PUSH 3
MUL
SUB
```

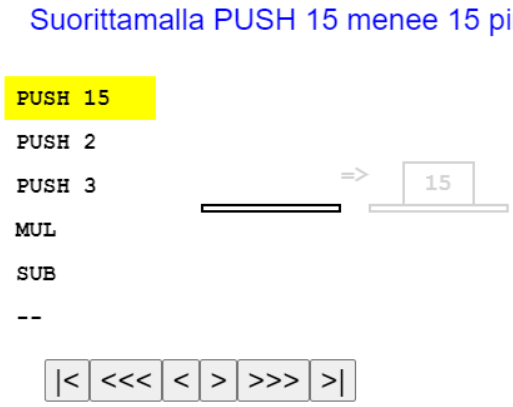
Alla olevalla simulaattorilla voit katsoa, miten pinolaskin suorittaa tämän käskyjonon. Nuolipainikkeilla voit simuloida toimintaa seuraavasti:

- [>]: ota askel

- >>>: käy animoidusti läpi koko loppuohjelma
- >|: aja suoraan koko ohjelma loppuun.

Painikkeet <, <<<, ja |< toimivat vastaavasti taaksepäin.

Simulaattorissa seuraavaksi suoritusvuorossa oleva käsky on korostettu. Kuvassa on nuolen vasemmalla puolella pinon tilanne ennen korostetun käskyn suorittamista ja oikealla puolella himmeämpänä tilanne käskyn suorittamisen jälkeen. Alussa pino on tyhjä.



Peruslaskutoimitusten laskemisessa pinolaskin eroaa perinteisestä laskimesta kahdella tavalla. Ensinnäkin luvut ja laskutoimitukset esitetään toisessa järjestyksessä. Toisekseen pinolaskimessa jokaisen käskyn jälkeen pinossa välittömästi tapahtuu jotain, kun taas perinteisessä laskimessa mitään ei tapahdu, ennen kuin koko suoritettava laskutoimitus on syötetty. Kummallakin tavalla on etunsa laskimen käyttömukavuuden kannalta.

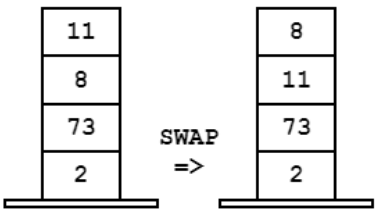
Haluamme seuraavaksi laajentaa pinolaskimemme sellaiseksi, että sille voidaan kirjoittaa *ohjelmia*, toisin sanoen käskyjonoja, joilla saadaan suoritetuksi mille tahansa annetuille luvuille jokin haluttu laskutoimitus. Tätä varten otamme käyttöön joukon käskyjä, joilla pinoa voidaan muokata muutenkin kuin päällimmäisen alkion osalta:

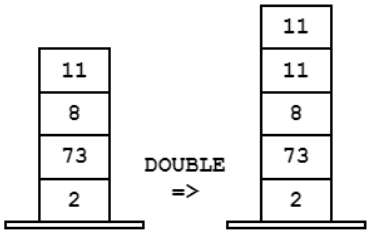
- `SWAP`: vaihda pinon kaksi päällimmäistä lukua keskenään
- `DOUBLE`: tee pinon päällimmäisestä luvusta kopio ja lisää se pinon päälle
- `POP`: poista pinon päällimmäinen luku
- `UP`: kierrä pinon sisältöä ylöspäin niin, että päällimmäinen luku siirtyy pinon pohjalle ja muut nousevat yhden askelen ylemmäs
- `DOWN`: kierrä pinon sisältöä alaspäin niin, että pohjimmainen luku siirtyy pinon päälle ja muut painuvat yhden askelen alemmas.

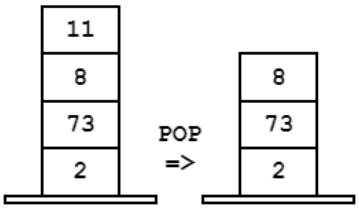
Käsky `SWAP` edellyttää, että pinossa on vähintään kaksi lukua, ja käskyt `DOUBLE`, `POP`, `UP` ja `DOWN`, että pinossa on ainakin yksi luku. Muuten kone menee virhetilaan ja sen toiminta päättyy.

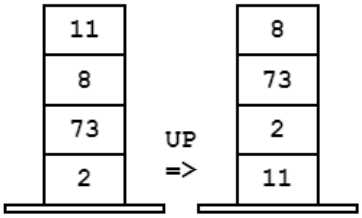
Normaalisti ohjelman suoritus päättyy, kun ohjelman viimeinen rivi on suoritettu.

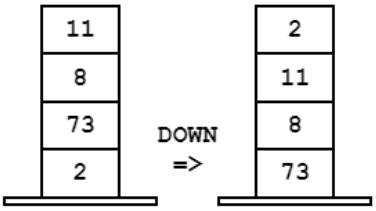
Seuraavat esimerkit havainnollistavat näitä käskyjä:











Yhdistelemällä näitä toimintoja voimme nyt kirjoittaa ohjelmia, jotka suorittavat jonkin monimutkaisen toimenpiteen, joka käsittelee useita lähtöarvoina annettuja lukuja. Kutsumme näitä lähtöarvoja ohjelman *syötteeksi* ja oletamme, että ne on tallennettu pinoon ennen ohjelman suorittamista. Tällaista ohjelmoitavaa pinolaskinta kutsumme *pinokoneeksi*.

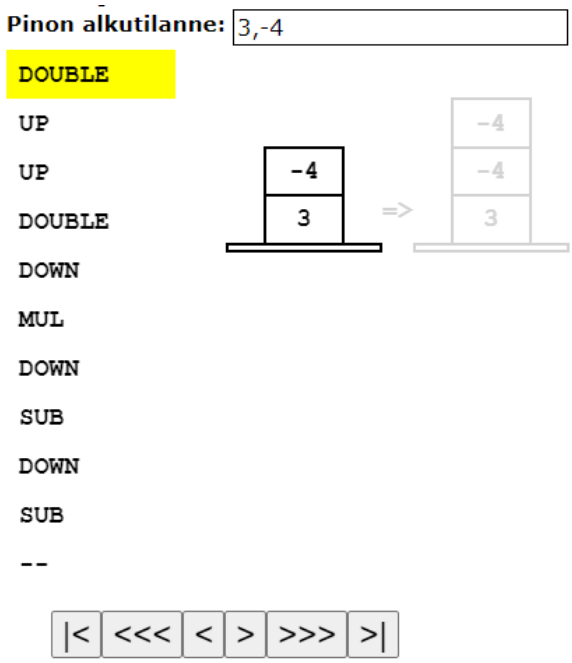
Esimerkiksi seuraava ohjelma laskee arvon lausekkeelle $x \cdot y - x - y$, kun syötearvot x ja y ovat pinossa ohjelman suorituksen alkaessa. Oletamme, että alkutilanteessa pinon päällä on y ja sen alla x eikä pinossa ole muita lukuja; tämä vastaa sitä, että aluksi tyhjään pinoon on tehty `PUSH` x ja `PUSH` y.

```
DOUBLE
UP
UP
DOUBLE
DOWN
MUL
DOWN
SUB
DOWN
SUB
```

Kokeile tämän ohjelman toimintaa eri syötteillä käyttämällä alla annettua simulaattoria. Laatikkoon **Pinon alkutilanne** voit kirjoittaa syötteen, jolla ohjelma suoritetaan. Pinon alkutilanne esitetään tässä ja jatkossa niin, että ensin mainitaan pinon pohjalla oleva luku ja viimeisenä

pinon päällä oleva luku. Luvut erotetaan toisistaan pilkulla. Siis esimerkiksi pinon alkutilanne `3,-4` vastaa tilannetta, joka syntyisi suorittamalla `PUSH 3` ja `PUSH -4` tässä järjestyksessä.

Ohjelman perusajatus on, että pinon päällä alun perin olevista kahdesta luvusta tehdään kopio pinon pohjalle käyttäen `DOUBLE`- ja `UP`-käskyjä. Kun kertolasku on suoritettu, alkuperäiset luvut saadaan `DOWN`-käskyillä takaisin pinon päälle vähennyslaskuja varten.



Edellisissä esimerkeissä ohjelma suoritti aina sama käskyt samassa järjestyksessä syötteen arvoista riippumatta, ja jokainen ohjelman käsky suoritettiin tasan kerran. Monimutkaisempia rakenteita varteen pinokoneessamme on *hyppykäskyjä*, joilla ohjelman käskyjen suoritusjärjestystä voi muuttaa. Pinokoneen hyppykäskyissä käytetään *nimiöitä* (englanniksi label) osoittamaan ohjelman kohtia, joihin voidaan hypätä. Nimiöinä käytetään tässä kirjaimia A-Z.

Hyppyihin liittyvät seuraavat käskyt:

- `LABEL` x, missä x on jokin kirjaimista `A-Z`: tämä ohjelmarivi itsessään ei tee mitään, mutta se merkitsee kohdan, johon voidaan hypätä muualta ohjelmasta
- `JUMP` x: ohjelmasta etsitään rivi `LABEL` x, ja suoritus jatkuu tästä kohdasta
- `JZERO` x: kuten `JUMP` x, mutta hyppy tehdään vain, jos pinon ylin luku on nolla; muussa tapauksessa suoritus jatkuu normaalisti
- `JPOS` x: kuten `JUMP` x, mutta hyppy tehdään vain, jos pinon ylin luku on positiivinen (nollaa aidosti suurempi); muussa tapauksessa suoritus jatkuu normaalisti
- `JGTR` x: kuten `JUMP` x, mutta hyppy tehdään vain, jos pinon ylin luku on suurempi kuin toiseksi ylin; muussa tapauksessa suoritus jatkuu normaalisti.

Mikään nimiö (kirjain `A-Z`) ei saa esiintyä samassa ohjelmassa usealla eri `LABEL`-rivillä. Hyppykäskyissä saa käyttää vain nimiöitä, joita vastaava `LABEL`-rivi on ohjelmassa. `LABEL`-rivi voi olla myös ohjelman viimeisenä, jolloin siihen hyppääminen tarkoittaa ohjelman päättymistä.

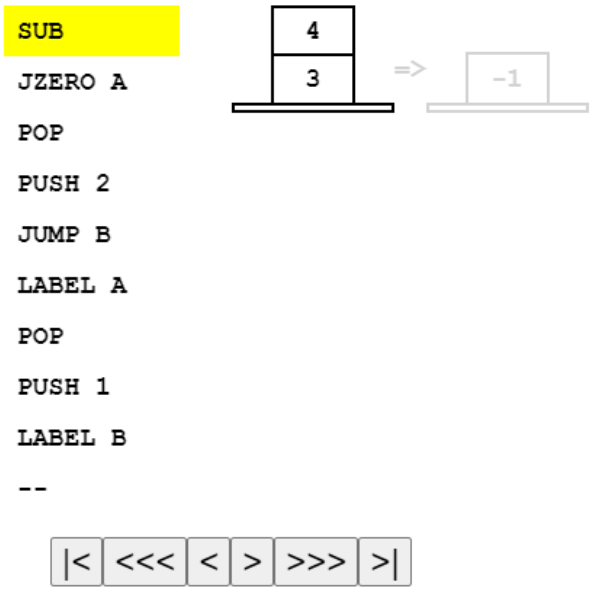
Käskyt `JZERO` ja `JPOS` edellyttävät, että pinossa on ainakin yksi luku, ja käsky `JGTR`, että pinossa on vähintään kaksi lukua. Muuten kone menee virhetilaan ja sen toiminta päättyy.

Hyppykäskyt eivät muuta pinon sisältöä.

Esimerkiksi seuraava ohjelma laittaa pinon päälle luvun 1, jos pinon kaksi ylintä lukua ovat yhtäsuuret, ja 2 muuten. Alkuperäiset luvut poistuvat pinosta:

```
SUB
JZERO A
POP
PUSH 2
JUMP B
LABEL A
POP
PUSH 1
LABEL B
```

Jos lukujen vähennyslaskun tulos on 0 eli luvut ovat yhtäsuuret, hypätään nimiöön `A`, jossa syntynyt 0 poistetaan `POP`-käskyllä ja pinon päälle asetetaan 1. Jos luvut ovat erisuuret, vähennyslaskun tulos ei ole nolla ja `JZERO` ei hyppää. Siis tässä tapauksessa suoritus jatkuu kohtaan, jossa pinon päälle asetetaan 2. Sen jälkeen pitää kuitenkin vielä hypätä ohjelman loppukohtaa osoittavaan nimiöön `B`, jotta tässä tapauksessa ohitetaan toiseen tapaukseen liittyvä `PUSH 1`.



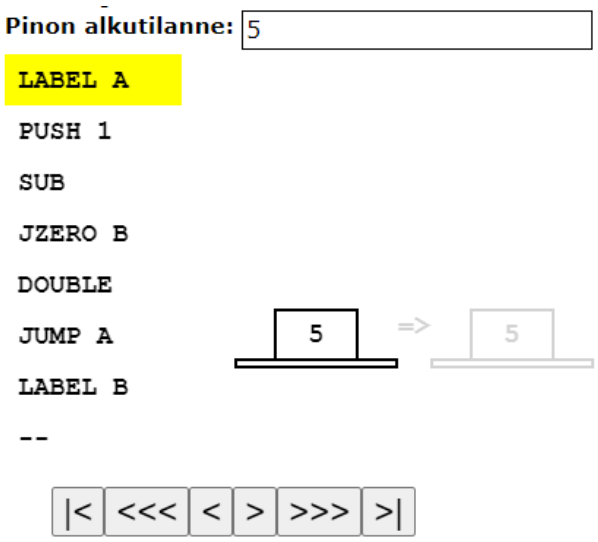
Edellisessä esimerkissä hyppykäskyjä käytettiin vain valitsemaan syötteen perusteella, mitkä ohjelmarivit suoritetaan ja mitkä jätetään suorittamatta. Hyppykäskyjen avulla voidaan muodostaa myös *silmukoita*, jolloin sama ohjelmarivi voidaan suorittaa useita kertoja tiettyä syötettä käsiteltäessä.

Esimerkkinä silmukasta tarkastellaan tehtävää, jossa syötteenä on annettu kokonaisluku n, ja ohjelman pitää tuottaa pinon sisällöksi $n-1, n-2, \ldots, 0$ (lueteltuna taas pohjalta alkaen, siis luku 0 pinon päällä). Syötteen n oletetaan olevan positiivinen (nollaa suurempi) kokonaisluku. Tehtävä voidaan ratkaista seuraavalla ohjelmalla:

```
LABEL A
PUSH 1
SUB
JZERO B
DOUBLE
JUMP A
LABEL B
```

Pinon päällä olevasta luvusta vähennetään 1 ja katsotaan, onko se nolla. Jos on, pinoon ei tarvita enää lisää lukuja ja hypätään ohjelman loppuun. Muuten saatu tulos otetaan talteen `DOUBLE`-käskyllä ja palataan kohtaan, jossa taas vähennetään 1. Siis pinon ylimmäksi luvuksi

tulee suorituksen aikana vuorollaan kukin luvuista $n, n-1, n-2, \ldots, 0$, minkä jälkeen ohjelma pysähtyy. Kokeile ohjelmaa pinokonesimulaattorilla jollain pienellä syötteellä (esim. 5) nähdäksesi, miten silmukka toimii.



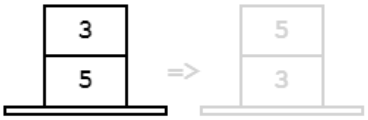
Monimutkaisemmissa tehtävissä voidaan joutua tekemään silmukassa enemmän toimenpiteitä pinolle kaikkien tarvittavien välitulosten pitämiseksi saatavilla. Esimerkiksi seuraava ohjelma laskee potenssin x^n (eli tulon, jossa luku x toistuu n kertaa), kun syötteenä on pinon päällä n ja sen alla x. Syötteiden x ja n oletetaan olevan positiivisia kokonaislukuja:

```
UP
PUSH 1
DOWN
LABEL A
JZERO B
PUSH 1
SUB
DOWN
DOUBLE
DOWN
MUL
UP
UP
JUMP A
LABEL B
POP
SWAP
POP
```

Alla on taas simulaattori, jolla voit kokeilla ohjelman toimintaa, ja sen alla on selitetty vaiheittain ohjelman toimintaperiaate. Perusajatuksena on, että pinossa säilytetään tietoa kolmesta eri asiasta. Luku x, jolle tehdään toistuvasti kertolaskuja, tarvitaan sellaisenaan. Lisäksi pidetään kirjaa siitä, kuinka monta kertolaskua on vielä suorittamatta ja mikä on tähän mennessä suoritettujen kertolaskujen tulos.

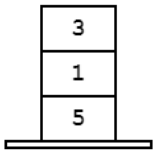
Pinon alkutilanne: 5,3

```
UP
PUSH 1
DOWN
LABEL A
JZERO B
PUSH 1
SUB
DOWN
DOUBLE
DOWN
MUL
UP
UP
JUMP A
LABEL B
POP
SWAP
POP
--
```

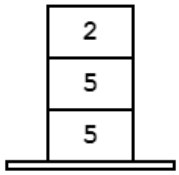


|< <<< < > >>> >|

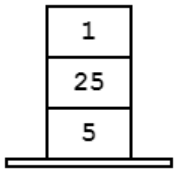
Tarkastellaan esimerkkinä potenssin $5^3 = 125$ laskemista. Siis aluksi pinon päällä on 3 ja sen alla 5. Ensimmäiset kolme riviä (ennen riviä `LABEL A`) lisäävät pinoon arvon 1, minkä jälkeen tilanne on seuraava:



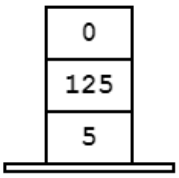
Siis kerrottava luku 5 on pinon pohjalla, tähän mennessä (kun ei ole suoritettu yhtään kertolaskua) välitulos on 1 (eli 5^0), ja suorittamatta on 3 kertolaskua. Koska pinon ylin luku eli suorittamatta olevien kertolaskujen määrä on nollasta poikkeava, `JZERO B` ei tee hyppyä. Pinon päällä olevasta laskurista vähennetään 1, minkä jälkeen välitulos 1 kerrotaan pinon pohjalta löytyvällä luvulla 5; tämä edellyttää hieman pinon pyörittelyä kerrottavien lukujen saamiseksi pinon päälle ja tuloksen saamiseksi omalle paikalleen. Kun tullaan riville `JUMP A`, pinon tilanne on



Siis kun suorittamatta on 2 kertolaskua, välitulos on 5; kerrottava luku pysyy koko ajan arvossa 5. Nyt hypätään takaisin kohtaan `LABEL A`, ja koska pinon päällä ei edelleenkään ole 0, suoritetaan samat operaatiot uudestaan. Tuloksena saadaan pinoon



eli välitulos on kasvanut arvoon 25 ja suorittamattomien kertolaskujen määrä on pienentynyt arvoon 1. Edelleen pinon päällä ei ole 0, joten suoritetaan vielä yksi kierros, minkä jälkeen pinossa on



Nyt pinon päällä on 0, joten `JZERO B` hyppää ohjelman loppuun, missä vielä siistitään pinosta pois muut luvut kuin haluttu lopputulos 125.

sec $-head$

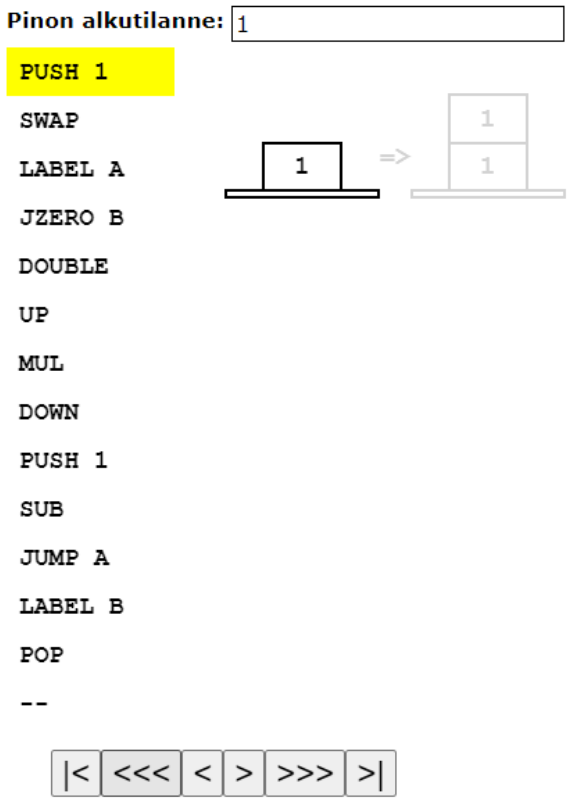
Tehtävä 3, kysymykset

Jokaisessa kysymyksessä tallenna vastauksesi “Tallenna”-painikkeella. Voit myöhemmin vaihtaa vastaustasi tallentamalla uuden vastauksen. Viimeisenä tallennettu vastaus jää voimaan.

Kysymys 3.1 (1 piste)

Alla on simulaattoriin valmiiksi kirjoitettu ohjelma, joka olettaa, että alkutilanteessa pinossa on yksi positiivinen kokonaisluku. Mitä tämä ohjelma tekee? Toisin sanoen kun alkutilanteessa pinon päällä on positiivinen kokonaisluku n ja pino on muuten tyhjä, niin mikä luku pinossa on ohjelman suorituksen päätyttyä? Valitse oikea vaihtoehto simulaattorin alla annetusta monivalinnasta.

Voit käyttää simulaattoria apuna kysymyksen ratkaisemisessa. Annetuista vastausvaihtoehdoista tasan yksi on oikea.



Valitse alla olevista oikea vaihtoehto

- ○ Ohjelma laskee annetulle luvulle n aritmeettisen summan 1:stä n:ään, eli $1 + 2 + 3 + ... + n$.
- ○ Ohjelma laskee annetulle luvulle n neliösumman 1:stä n:ään, eli $1^2 + 2^2 + 3^2 + ... + n^2$.
- ○ Ohjelma laskee annetun luvun n kertoman, eli $1 \cdot 2 \cdot 3 \cdot ... \cdot n$.
- ○ Ohjelma laskee annetulle luvulle n tulon luvuista 1:stä n:ään korotettu potenssiin itsensä kanssa, eli $1^1 \cdot 2^2 \cdot 3^3 \cdot ... \cdot n^n$.

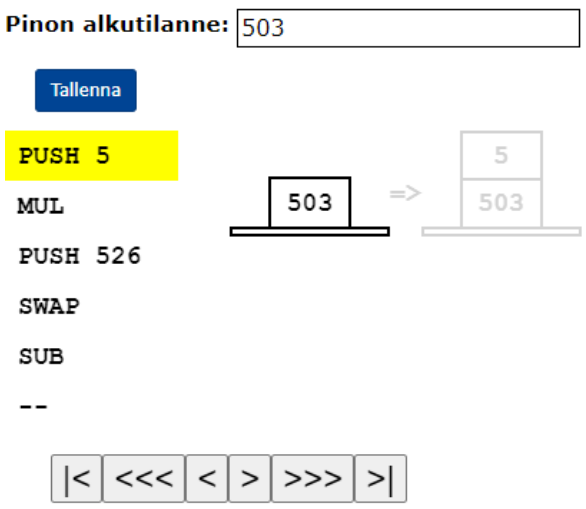
Kysymys 3.2 (1 piste)

Alla on simulaattoriin valmiiksi kirjoitettu ohjelma, joka olettaa, että alkutilanteessa pinossa on yksi positiivinen kokonaisluku. Mikä pitää olla pinon alkutilanne, jotta ohjelman suorittamisen jälkeen pinossa olisi luku 411 eikä mitään muuta?

Esimerkiksi jos pinossa olisi aluksi 5, niin ohjelman suorituksen jälkeen pinossa olisi 501 eikä 411, joten “5” ei ole oikea vastaus tähän kysymykseen.

Voit halutessasi kokeilla simulaattorilla ohjelman toimintaa eri alkutilanteilla. Kun olet mielestäsi löytänyt oikean alkutilanteen, kirjoita se alkutilanneruutuun ja paina “Tallenna”-painiketta, jolloin se tallentuu vastaukseksesi. Voit myöhemmin vaihtaa vastaustasi; viimeisenä tallentamasi alkutilanne jää voimaan.

Simulaattori:



Ohje kysymyksiin 3.3 3.10

Kysymyksissä 3.3 3.10 vastauksena on pinokoneen ohjelma. Sinun pitää kirjoittaa vastauksesi kysymyksessä annetun simulaattorin asianmukaiseen kohtaan. Voit kirjoittaa käskyt itse tai käyttää käskyille annettuja painikkeita.

Lopuksi tallenna vastauksesi käyttämällä "Tallenna"-painiketta. Jos tallennat useita vastauksia, viimeisenä tallennettu jää voimaan. Aina kun painat "Tallenna"-painiketta, simulaattori testaa ohjelmasi toimintaa muutamalla esimerkkisyötteellä ja antaa palautetta siitä, toimiko ohjelmasi oikein.

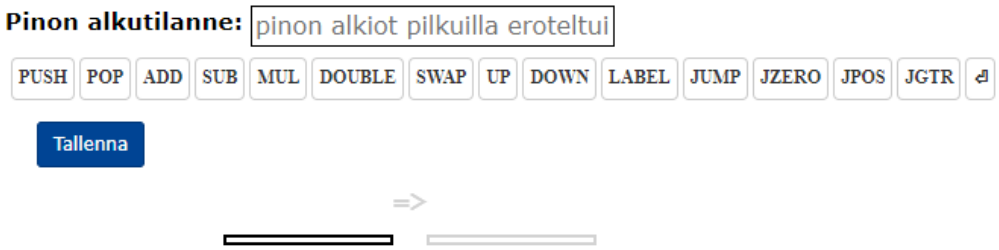
Kaikissa tehtävissä on vaatimuksena, että ohjelmasi suoritus pysähtyy normaalisti (ohjelman viimeinen rivi suoritetaan) eikä ohjelmasi aiheuta virhetilanteita (esimerkiksi yrittämällä suorittaa `ADD`-käskyn, kun pinossa on vain yksi luku).

Voit halutessasi käyttää simulaattoria ohjelmasi kokeilemiseen erilaisilla alkutilanteilla. Tämä ei vaikuta vastauksesi arvosteluun. Joissain kysymyksissä syötteenä annetut luvut voivat vaikuttaa ohjelmasi suorittamien askelten lukumäärään ja pinon kokoon. Pidä siksi kokeilusyötteittesi luvut kohtuullisen pieninä (pienempinä kuin 20), jotta simulaattori pystyy näyttämään suorituksen kunnolla.

Kysymys 3.3 (2 pistettä)

Kirjoita pinokoneelle ohjelma, joka laskee annetulle kokonaisluvulle x lausekkeen $6 \cdot x - 5$ arvon. Ohjelman suorituksen alkaessa oletetaan, että pinossa on vain yksi kokonaisluku x, ja suorituksen päättyessä pinossa tulee olla vain annetun lausekkeen arvo tälle luvulle x.

Esimerkiksi jos alkutilanteessa pinossa on vain luku 2, niin ohjelman suorituksen jälkeen pinossa tulee olla ainoastaan luku 7.

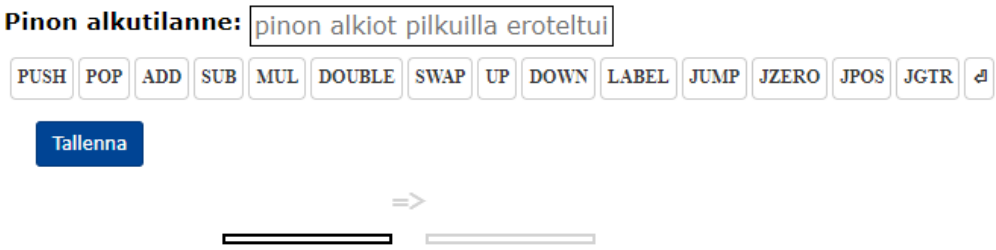


Kysymys 3.4 (2 pistettä)

Kirjoita pinokoneelle ohjelma, joka laskee annetuille kokonaisluvuille x ja y lausekkeen $x^2 - x \cdot y$. Ohjelman suorituksen alkaessa oletetaan, että pinossa on vain kaksi kokonaislukua x ja y, joista x pohjalla ja y päällä. Suorituksen päättyessä pinossa tulee olla vain annetun lausekkeen arvo näille luvuille x ja y.

Huomaa lukujen x ja y järjestys alkutilanteessa. Tämä vastaa sitä, että olisi suoritettu `PUSH` x ja `PUSH` y, tässä järjestyksessä.

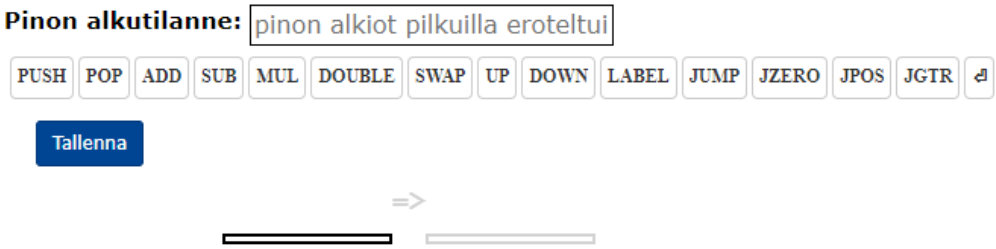
Jos esimerkiksi alkutilanne on $5, -2$ (pinossa on luku 5 pohjalla ja -2 päällä), niin ohjelman suorituksen jälkeen pinossa tulee olla ainoastaan luku 35.



Kysymys 3.5 (2 pistettä)

Kirjoita pinokoneelle ohjelma, joka laskee annetulle kokonaisluvulle x sen itseisarvon $|x|$. Ohjelman suorituksen alkaessa oletetaan, että pinossa on vain yksi kokonaisluku x, ja suorituksen päättyessä pinossa tulee olla vain tämän luvun itseisarvo.

Esimerkiksi jos alkutilanteessa pinossa on vain luku -51, niin ohjelman suorituksen jälkeen pinossa tulee olla ainoastaan luku 51.



Kysymys 3.6 (2 pistettä)

Kirjoita pinokoneelle ohjelma, joka laskee annetulle kokonaisluvulle x lausekkeen $16 \cdot x$ arvon. Ohjelman suorituksen alkaessa oletetaan, että pinossa on vain yksi kokonaisluku x, ja suorituksen päättyessä pinossa tulee olla vain luku $16 \cdot x$. Ohjelmassa **ei saa** käyttää käskyä MUL.

Täysien pisteiden saamiseksi ohjelmassa saa olla korkeintaan 8 käskyä. Pidemmästä mutta vaatimukset muuten täyttävästä ohjelmasta saa yhden pisteen.

Esimerkiksi jos alkutilanteessa pinossa on vain luku 3, niin ohjelman suorituksen jälkeen pinossa tulee olla ainoastaan luku 48.

Pinon alkutilanne: pinon alkiot pilkuilla eroteltui

PUSH POP ADD SUB MUL DOUBLE SWAP UP DOWN LABEL JUMP JZERO JPOS JGTR ⏎

Tallenna

=>

Kysymys 3.7 (2 pistettä)

Alkutilanteessa pinossa on kolme kokonaislukua. Kirjoita pinokoneelle ohjelma, joka kääntää lukujen järjestyksen päinvastaiseksi; toisin sanoen päällimmäisestä luvusta tulee pohjimmainen luku ja päinvastoin. Pinoon ei tule lisätä eikä siitä tule poistaa lukuja.

Esimerkiksi jos alkutilanne on $4, 2, 3$ (pinossa luku 4 pohjalla, sen päällä 2 ja päällimmäisenä 3), niin ohjelman suorituksen jälkeen pinon tilanteen tulee olla $3, 2, 4$ (siis 3 pohjalla ja 4 päällimmäisenä).

Pinon alkutilanne: pinon alkiot pilkuilla eroteltui

PUSH POP ADD SUB MUL DOUBLE SWAP UP DOWN LABEL JUMP JZERO JPOS JGTR ⏎

Tallenna

=>

Kysymys 3.8 (2 pistettä)

Alkutilanteessa pinossa on kolme kokonaislukua. Kirjoita pinokoneelle ohjelma, joka jättää pinoon ainoaksi luvuksi näistä pienimmän.

Esimerkiksi jos pinon alkutilanne on $4, 5, 1$, niin ohjelman suorituksen jälkeen pinossa tulee olla vain luku 1.

Pinon alkutilanne: pinon alkiot pilkuilla eroteltui

PUSH POP ADD SUB MUL DOUBLE SWAP UP DOWN LABEL JUMP JZERO JPOS JGTR ⏎

Tallenna

=>

Kysymys 3.9 (3 pistettä)

Alkutilanteessa pinossa on kaksi positiivista kokonaislukua n ja x, päällimmäisenä n. Kirjoita pinokoneelle ohjelma, joka tallentaa pinoon n kappaletta lukuja x. Ohjelman suorituksen jälkeen pinossa tulee olla ainoastaan n kappaletta alkuperäistä lukua x.

Esimerkiksi jos pinon alkutilanne on $5, 4$ (luku 5 pohjalla ja 4 päällä), niin ohjelman suorituksen jälkeen pinon tilanteen tulee olla $5, 5, 5, 5$.

Pinon alkutilanne: pinon alkiot pilkuilla eroteltui

PUSH POP ADD SUB MUL DOUBLE SWAP UP DOWN LABEL JUMP JZERO JPOS JGTR ⏎

Tallenna

=>

Kysymys 3.10 (3 pistettä)

Alkutilanteessa pinossa on positiivinen kokonaisluku. Kirjoita pinokoneelle ohjelma, joka lisää pinon päälle luvun

- luvun 1, jos pinossa alun perin ollut luku on parillinen
- luvun 0, jos pinossa alun perin ollut luku on pariton.

Ohjelman suorituksen jälkeen pinossa tulee olla tasan kaksi lukua: päällimmäisenä 0 tai 1 ja alimmaisena pinossa alun perin ollut luku.

Esimerkiksi jos alkutilanteessa pinossa on vain luku 5, niin ohjelman suorituksen jälkeen pinon tilanteen tulee olla $5, 0$ (pohjalla 5 ja päällä 0). Jos taas alkutilanteessa pinossa on vain luku 12, niin ohjelman suorituksen jälkeen pinon tilanteen tulee olla $12, 1$ (pohjalla 12 ja päällä 1).

Pinon alkutilanne: pinon alkiot pilkuilla eroteltui

PUSH POP ADD SUB MUL DOUBLE SWAP UP DOWN LABEL JUMP JZERO JPOS JGTR ⏎

Tallenna

=>