Aritmetikai kifejezések lengyelformára hozása

Készítették:

Santák Csaba és **Kovács Péter**, 2005

ELTE IK programtervező matematikus szak

Aritmetikai kifejezések kiértékelése

- Gyakran felmerülő programozási feladat, hogy egy adott aritmetikai kifejezést kell kiértékelni (kiszámolni).
- A probléma nehézségét az okozza, hogy a kifejezések megszokott, természetes formája (az ún. *infix* írásmód, amikor az egyes műveleti jelek az operandusaik között helyezkednek el) nem egyértelmű és algoritmikusan nehezen kezelhető.
- Az egyik legjobb és leghatékonyabb megoldási módszer az, hogy a kifejezést első lépésben átalakítjuk ún. *lengyelformává*, utána pedig azt értékeljük ki.

A lengyelforma

- **Łukasiewicz** lengyel matematikus kidolgozta az aritmetikai formulák egy olyan átalakítási módját, amelynek segítségével a kiértékelés rendkívül egyszerűen és gyorsan megvalósítható. Ennek egy módosított változata terjedt el az informatikában.
- Ebben az alakban a műveleti jelek az operandusaik után állnak, ezért *postfix* formának, illetve a szerző tiszteletére *lengyelformának* nevezik*. (Angol szakirodalomban: *reverse Polish notation,* RPN)

* Megjegyezzük, hogy egy kifejezés lengyelformája megegyezik a kifejezésfa postorder bejárásával.

Példák

- a+b → ab+
- a+b*c → abc*+
- a*b+(c-d)/2 → ab*cd-2/+
- a+b+c*d/2-e*f → ab+cd*2/+ef*-
- x:=a+b → xab+:=
- x:=a*b^2 → xab2^*:=
- …

A lengyelforma tulajdonságai

Foglaljuk össze a lengyelfoma legfontosabb jellemzőit:

- nincs benne zárójel,
- az operandusok sorrendje ugyanaz, mint az eredeti kifejezésben,
- a műveleti jelek sorrendje megegyezik a kifejezés helyes kiértékeléséhez szükséges elvégzési sorrendjükkel,
- a műveleti jeleket közvetlenül megelőzik az operandusai (erre utal a *postfix* elnevezés).

Ezek biztosítják, hogy egy lengyelformára hozott kifejezés kiértékelése könnyen elvégezhető legyen.

Aritmetikai kifejezések

Az algoritmus tárgyalása előtt foglaljuk össze a szükséges fogalmakat és szabályokat.

Műveletek precedenciái:

Ebben a tárgyalásban hat műveletet tekintünk megengedettnek: a négy alapműveletet, a hatványozást és az értékadást.

Ezek precedencia-szintjei a következők (a legmagasabbtól a legalacsonyabbig):

- ^ (hatványozás)
- *, / (szorzás, osztás)
- +, - (összeadás, kivonás)
- := (értékadás, definiáló egyenlőség*)

* Több programozási nyelvhez hasonlóan megengedettnek tekintjük a többszörös értékadásokat is (pl. a:=b:=0).

Aritmetikai kifejezések

Kifejezések kiértékelésekor figyelembe kell vennünk az egyes műveletek asszociativitási (zárójeleződési) szabályait is.

1. A négy alapművelet megegyezés szerint *balról jobbra* zárójeleződik:

   $$a+b+c \rightarrow ((a+b)+c)$$

2. A hatványozás és az értékadás megegyezés szerint *jobbról balra* zárójeleződik*:

   $$x\text{^}2\text{^}3 \rightarrow (x\text{^}(2\text{^}3)) = x\text{^}8$$

   (Ha balról jobbra zárójeleznénk, akkor x^6-t kapnánk.)

* Többszörös értékadás esetén a balról jobbra zárójeleződés nem is értelmezhető.
Pl. *a:=b:=1* → *(a:=b):=1* nem értelmezhető megfelelően, csak az *a:=(b:=1)*.

A lengyelformára hozás feladata

- Adott egy szintaktikusan helyes *infix* aritmetikai kifejezést tartalmazó szekvenciális input. Alakítsuk át lengyelformára.
- Tegyük fel, hogy az input elemei nem karakterek, hanem a kifejezés egységei (lexikális elemei): operandus (változó, konstans stb.), operátor vagy zárójel; és minden elemről egyértelműen el tudjuk dönteni, hogy ezek közül melyik.

Lengyelformára hozás

Az algoritmus során az inputot szekvenciálisan dolgozzuk fel, és egy (kezdetben üres) verem segítségével állítjuk elő a lengyelformát:

- Ha a kifejezésben a soron következő elem operandus, akkor azt kiírjuk a kimenetre (operandus tehát sosem kerül a verembe!).
- Ha nyitó zárójel következik, akkor azt a verembe dobjuk.
- Ha csukó zárójel következik, akkor sorban kivesszük a verem tetejéről az elemeket az első (legfelső) nyitó zárójelig, és kiírjuk őket a kimenetre, majd a nyitó zárójelet is kivesszük és eldobjuk (zárójelek tehát nem fognak megjelenni a kimeneten, és csukó zárójel sem kerül soha a verembe).

Lengyelformára hozás

- Ha operátor következik, akkor az első nála *alacsonyabb* (illetve jobbról balra zárójeleződő művelet esetén az első *nem magasabb*) precedenciájú operátorig, vagy az első nyitó zárójelig kivesszük a verem tetejéről az operátorokat, és kiírjuk a kimenetre. Végül a most olvasott operátort betesszük a verembe.
- Amikor az input végére érünk, a veremből az összes még benne lévő operátort kivesszük és kiírjuk a kimenetre.

Példa

a+b+c*d/2-e*f

Hozzuk lengyelformára az alábbi kifejezést:

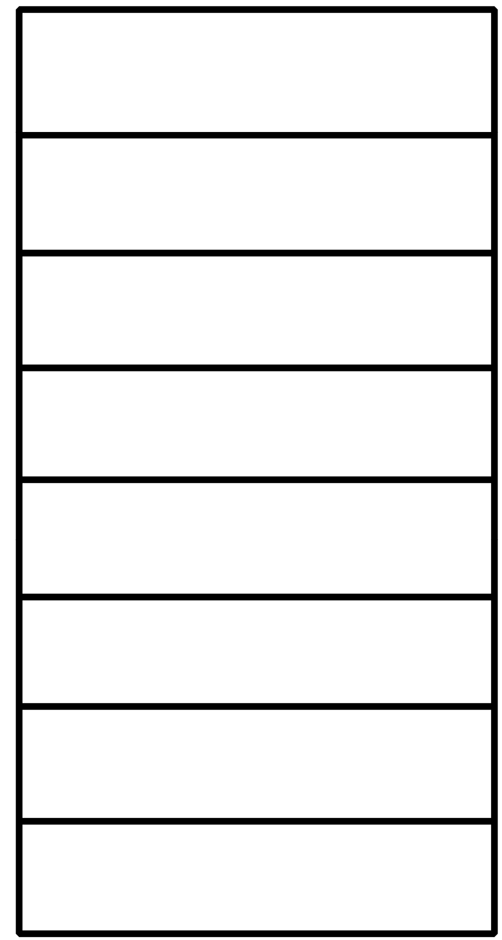
a+b+c*d/2-e*f

Adjuk meg lépésről lépésre a verem tartalmát és a kimenetet.

A verem kezdetben üres. Az első jel operandus, tehát kiírjuk a kimenetre.

Kimenet:

a



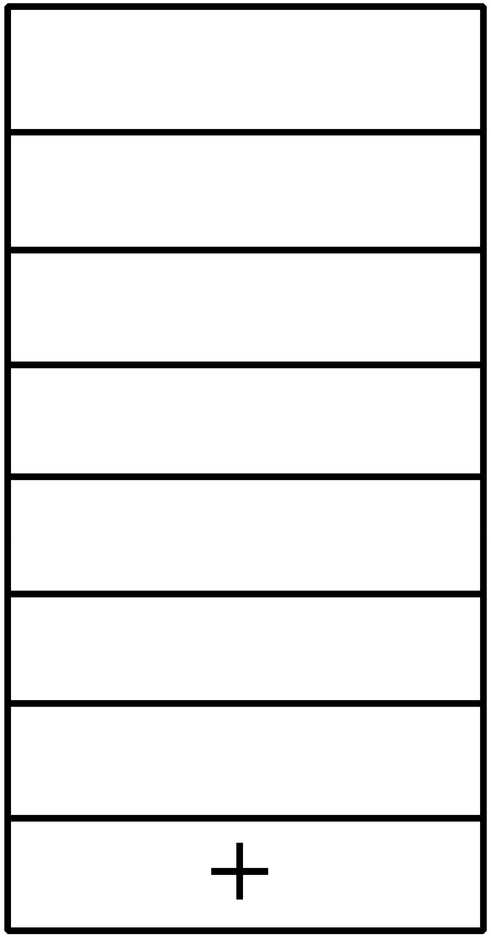
Példa

a+b+c*d/2-e*f

Operátor következik, tehát betesszük a verembe.

Kimenet:

a



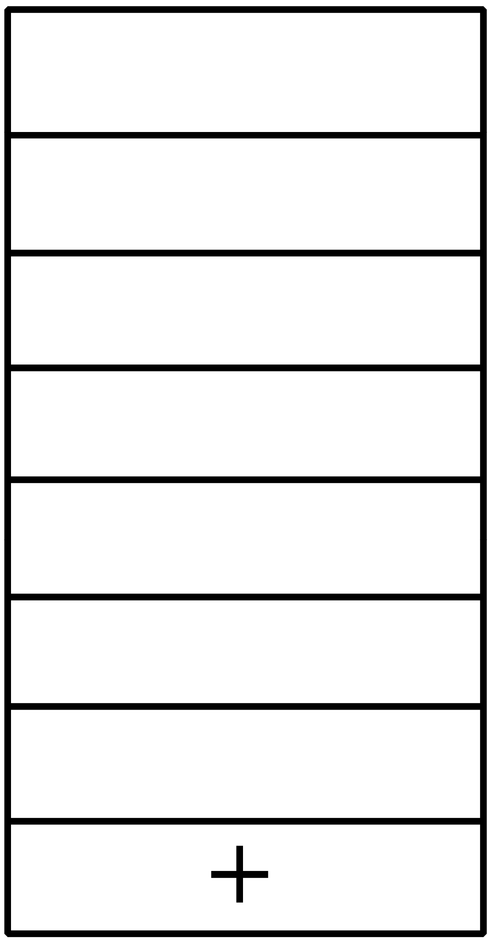
Példa

a+b+c*d/2-e*f

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

a b



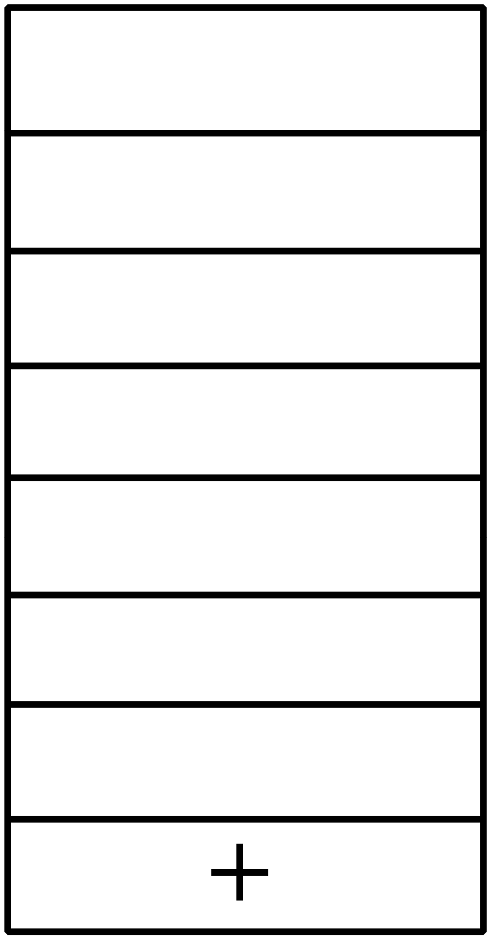
Példa

a+b+c*d/2-e*f

Operátor következik. Mivel vele azonos precedenciájú operátor van a veremben, így azt kiírjuk a kimenetre, az új „+” jelet pedig betesszük a verembe. (Az összeadás ugyanis balról jobbra zárójeleződik.)

Kimenet:

a b +



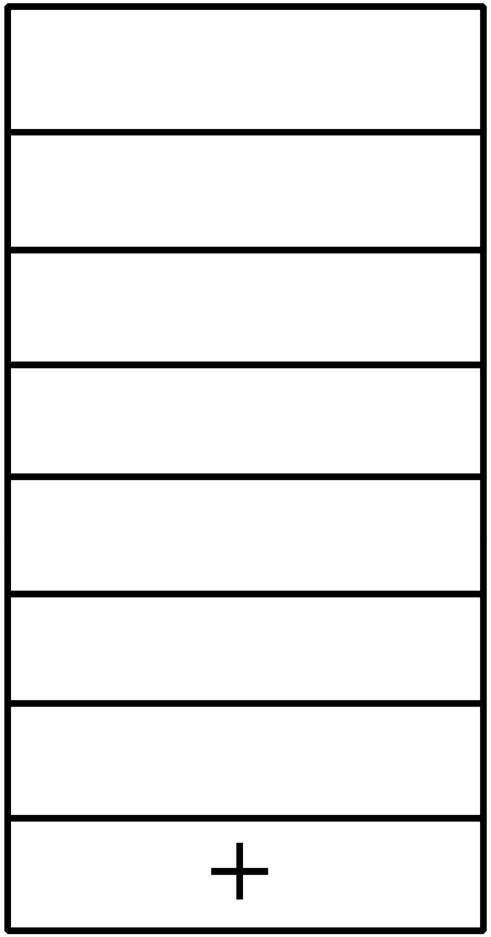
Példa

a+b+c*d/2-e*f

Operandus következik,
kiírjuk a kimenetre.

Kimenet:

a b + c



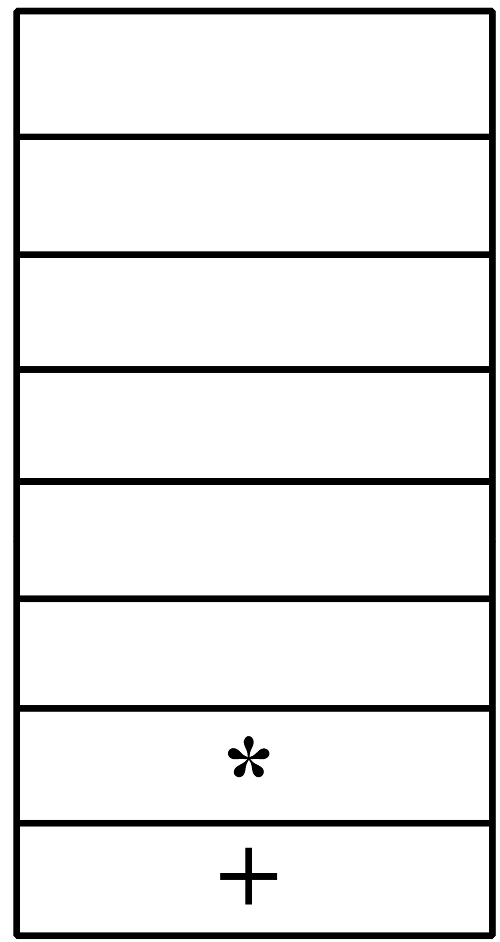
Példa

a+b+c*d/2-e*f

Operátor következik, ezért mivel magasabb precedenciájú a verem tetején lévő operátornál, betesszük a verembe.

Kimenet:

a b + c



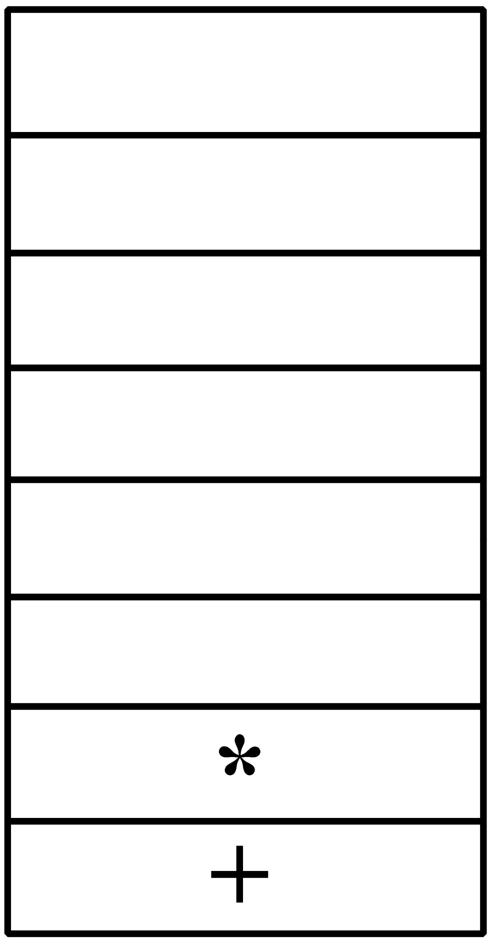
Példa

a+b+c*d/2-e*f

Operandus következik,
kiírjuk a kimenetre.

Kimenet:

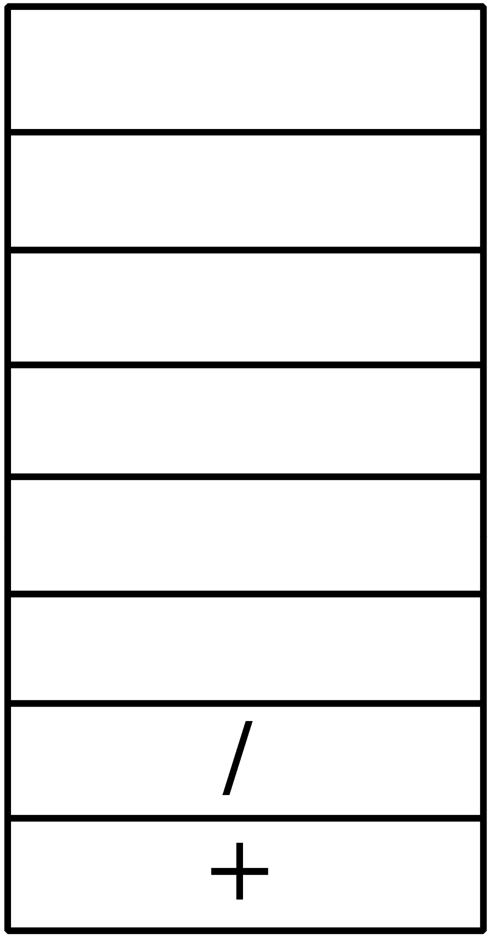
a b + c d



Példa

a+b+c*d/2-e*f

Operátor következik. Mivel vele azonos precedenciájú operátor van a veremben, így azt kiírjuk a kimenetre, a most olvasott „/” jelet pedig betesszük a verembe.



Kimenet:

a b + c d *

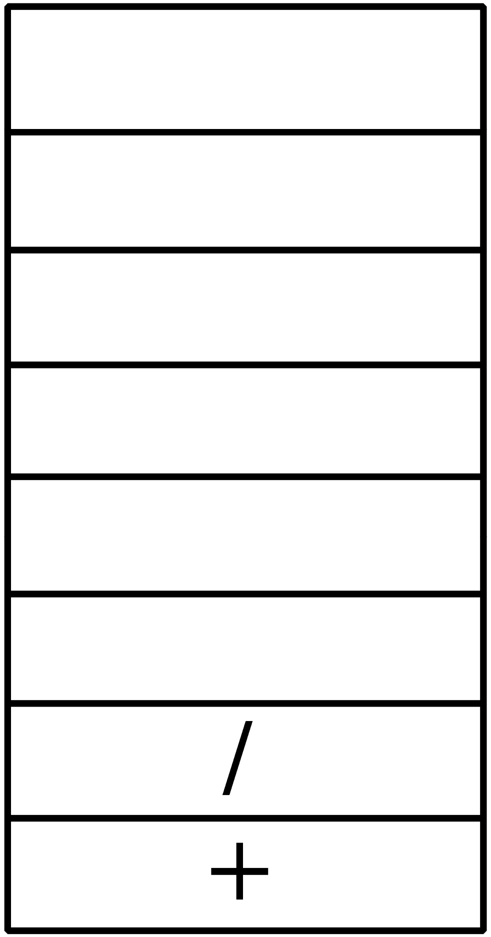
Példa

a+b+c*d/2-e*f

Operandus következik,
kiírjuk a kimenetre.

Kimenet:

a b + c d * 2



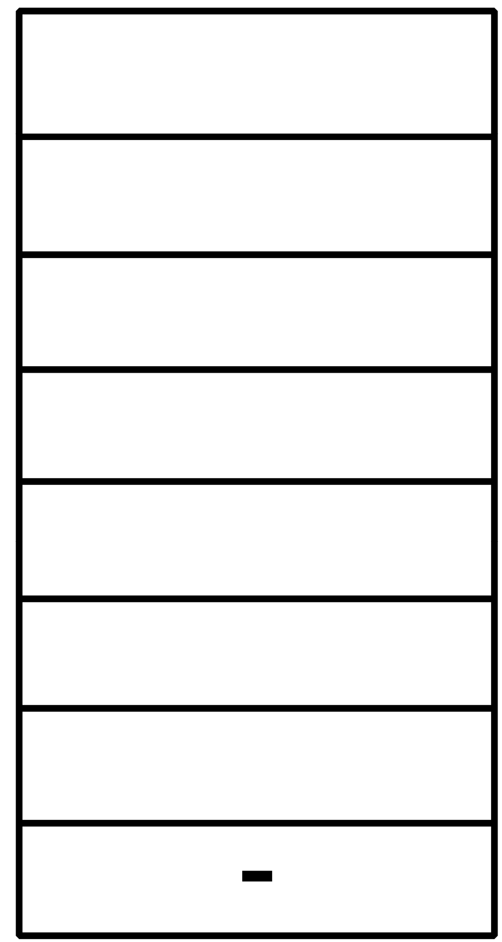
Példa

a+b+c*d/2-e*f

Operátor következik. Mivel a veremben egy nála magasabb, majd utána egy vele azonos precedenciájú operátor van, így azokat kiírjuk a kimenetre, a most olvasott „-” jelet pedig betesszük a verembe.

Kimenet:

a b + c d * 2 / +



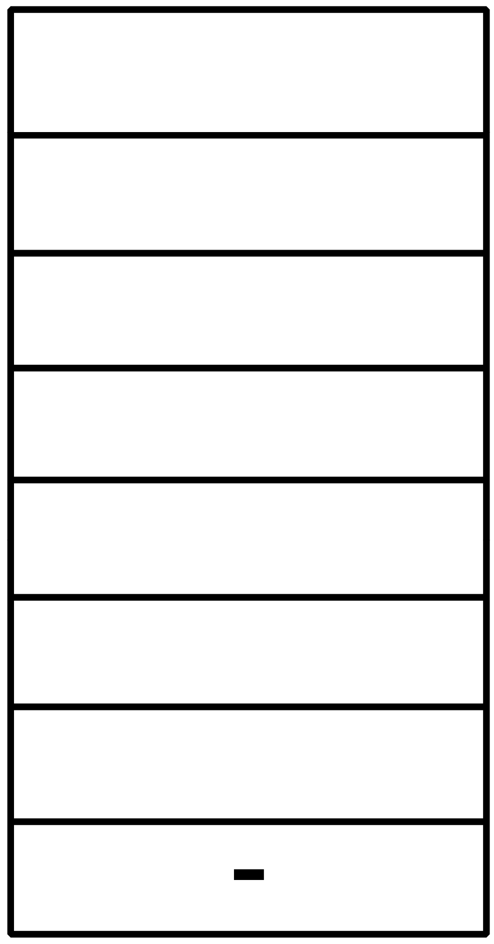
Példa

a+b+c*d/2-e*f

Operandus következik, kiírjuk a kimenetre.

Kimenet:

a b + c d * 2 / + e



Példa

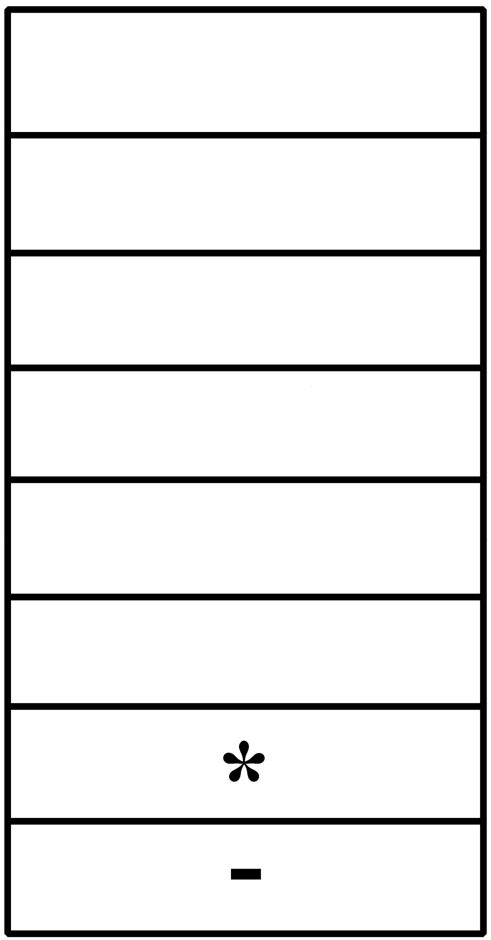
a+b+c*d/2-e*f

Operátor következik.

Betesszük a verembe, mivel magasabb precedenciájú, mint a verem tetején lévő operátor.

Kimenet:

a b + c d * 2 / + e



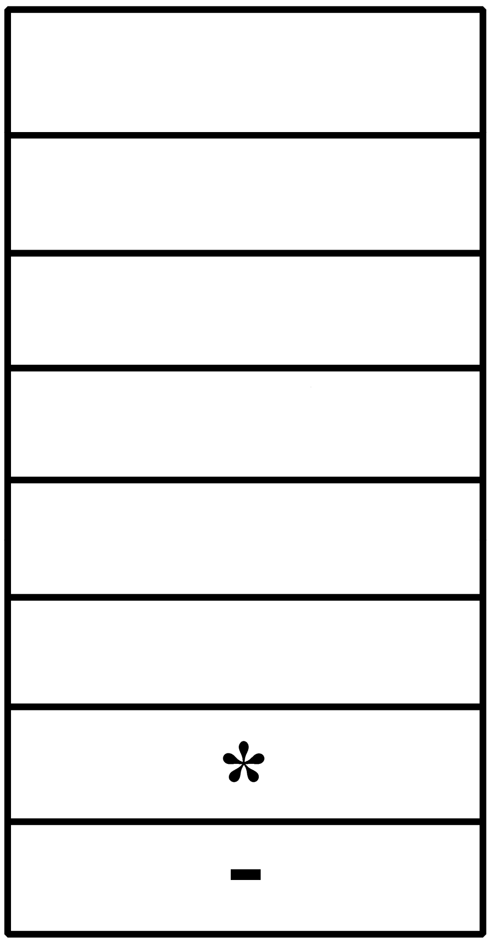
Példa

a+b+c*d/2-e*f

Operandus következik, kiírjuk a kimenetre.

Kimenet:

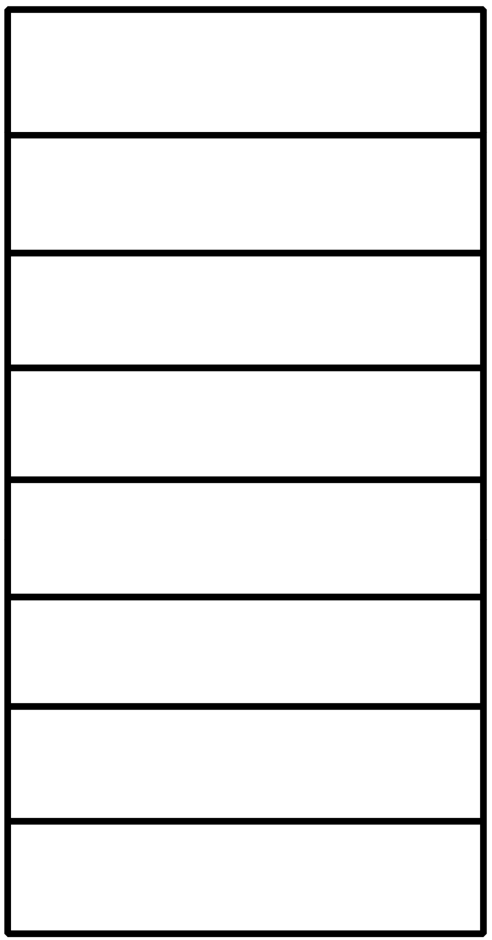
a b + c d * 2 / + e f



Példa

$a+b+c*d/2-e*f$

A kifejezés végére értünk, a veremben lévő összes operátort kivesszük és kiírjuk a kimenetre.



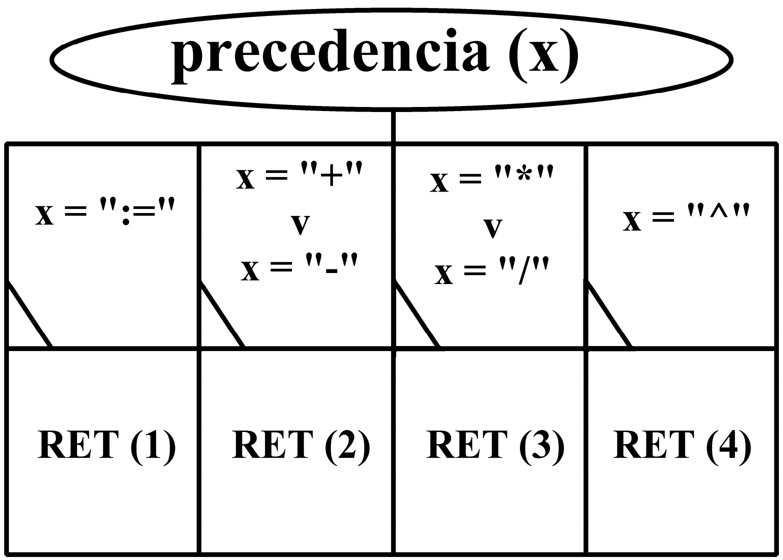
Kimenet:

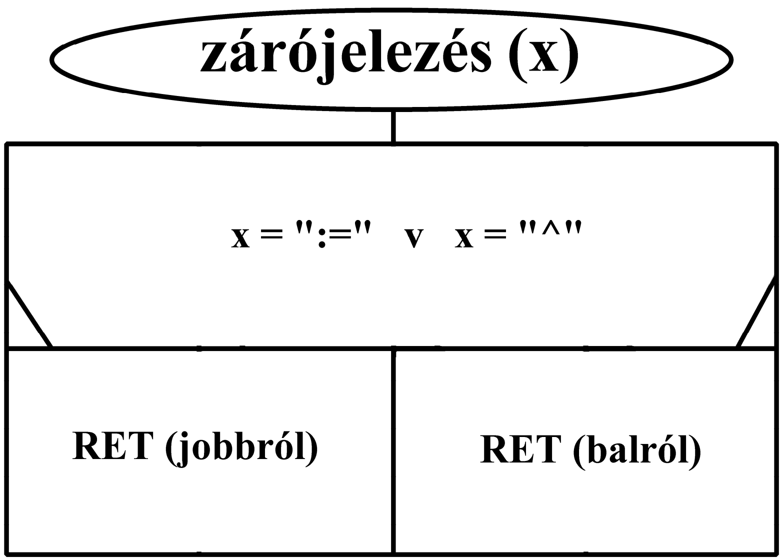
a b + c d * 2 / + e f * -

Az algoritmus struktogramja

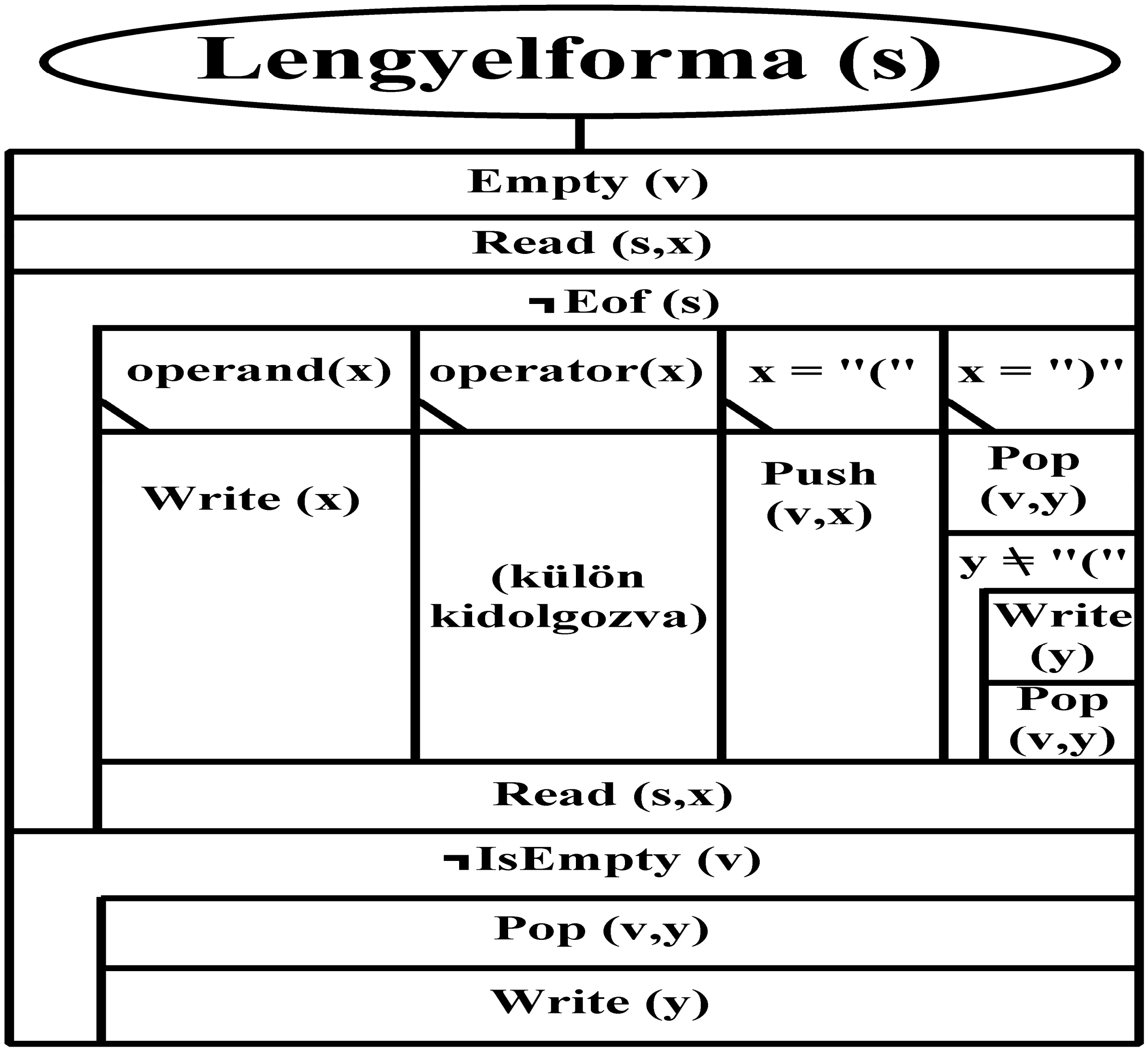
A korábbiakban megfogalmazott kikötések, egyszerűsítések mellett adjuk meg az algoritmus struktogramját is.

Ehhez először is vezessünk be két segédfüggvényt, amelyek megadják egy operátor precedenciáját és zárójeleződésének irányát:

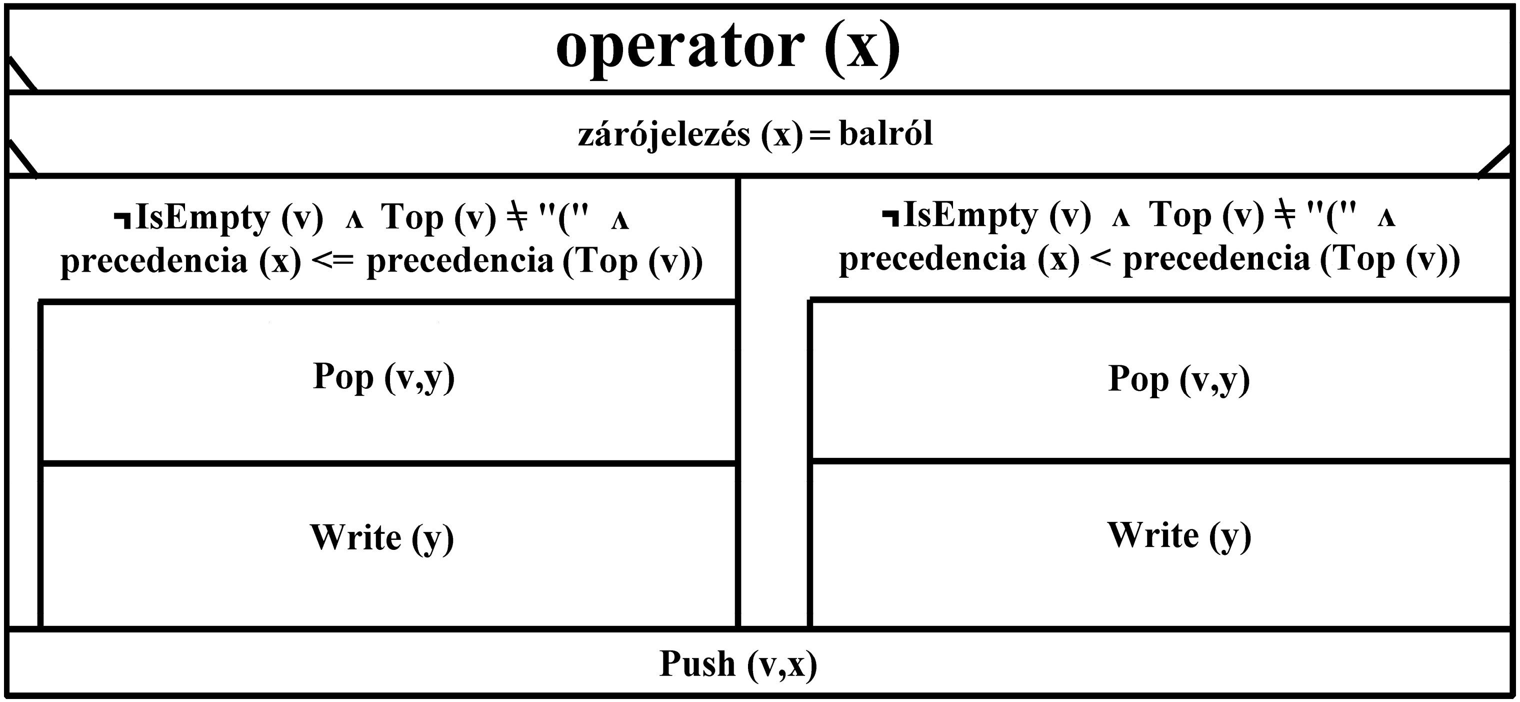




Az algoritmus struktogramja



Az algoritmus struktogramja

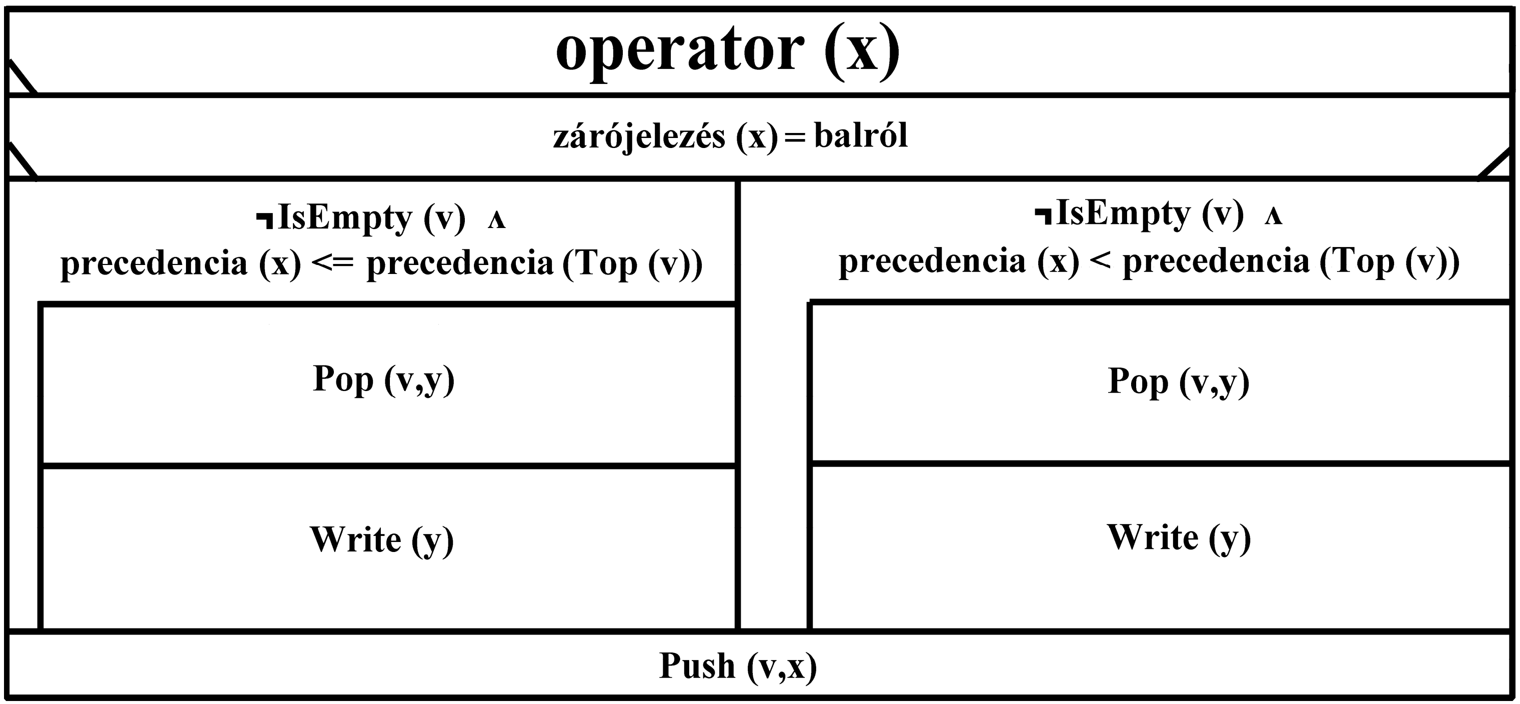


Megjegyzés: A fenti elágazás két (szerkezetileg azonos) ága összevonható, ha egy bonyolultabb ciklusfeltételt írunk fel:

$\neg$IsEmpty(v) $\wedge$ Top(v) $\neq$ "(" $\wedge$
((zárójelezés(x) = balról $\wedge$ precednecia(x) <= precedencia(Top(v)))
$\vee$ (zárójelezés(x) = jobbról $\wedge$ precednecia(x) < precedencia(Top(v))) **)**

Az algoritmus struktogramja

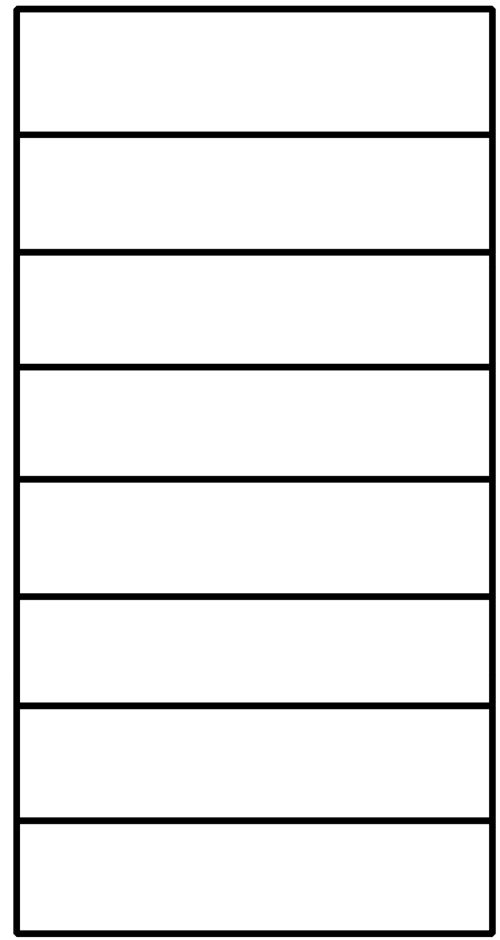
Ha a precedencia függvényünket a nyitó zárójelre is értelmezzük oly módon, hogy annak a precedenciája legyen a legalacsonyabb (nulla), akkor az *operator(x)* ágban szereplő ciklusok feltételei leegyszerűsödnek:



További példák

1. x:=a+b^3^2

Hozzuk lengyelformára a fenti kifejezést, és nézzük meg minden lépésnél a verem tartalmát, illetve a lengyelforma kimenetének aktuális állását.



A verem kezdetben üres.

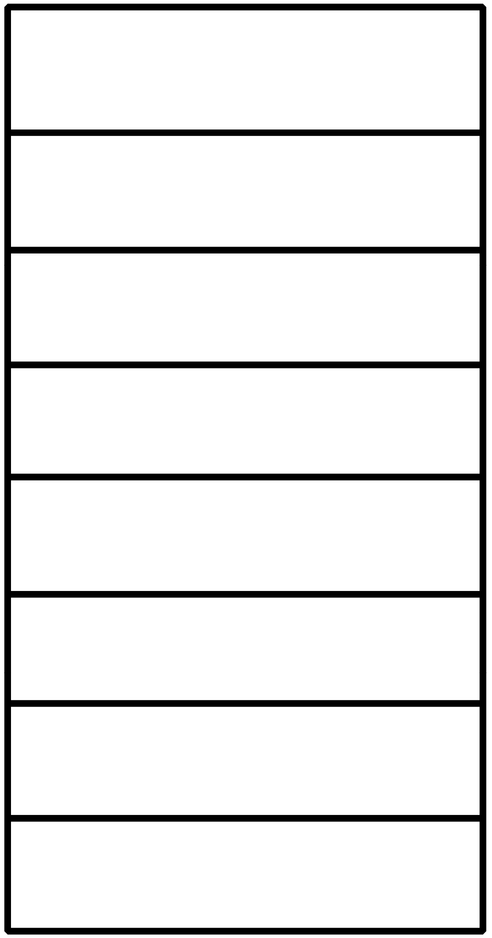
1. x:=a+b^3^2

Operandus jön, tehát kiírjuk
a kimenetre.

A verem még mindig üres.

Kimenet:

x

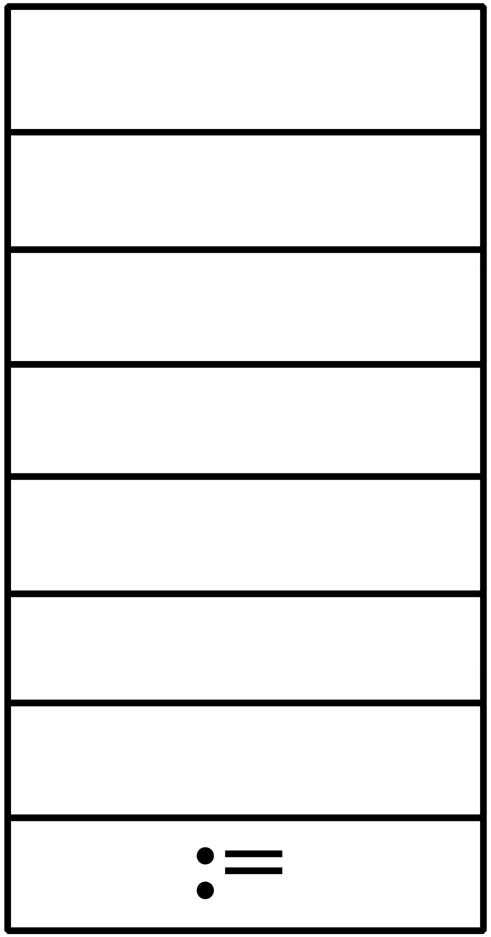


1. x:=a+b^3^2

Operátor jön, berakjuk a verembe.

Kimenet:

x

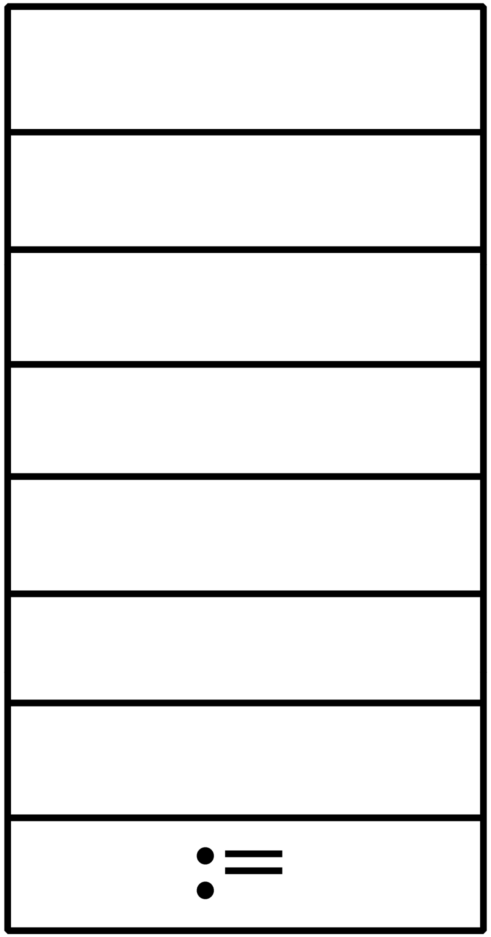


1. x:=a+b^3^2

Operandus jön, tehát kiírjuk a kimenetre.

Kimenet:

x a

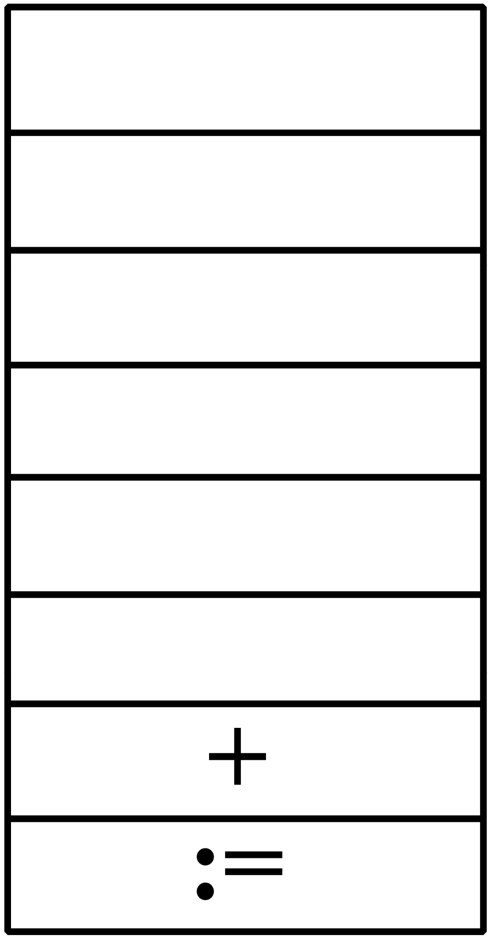


1. x:=a+b^3^2

Operátor jön, berakjuk a verembe a szabályoknak megfelelően.

Kimenet:

x a

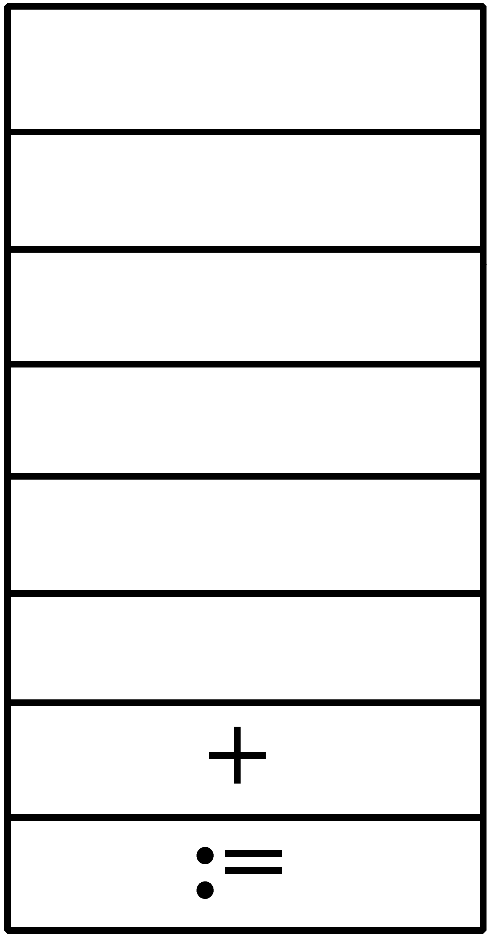


1. x:=a+b^3^2

Operandus jön, tehát kiírjuk a kimenetre.

Kimenet:

x a b

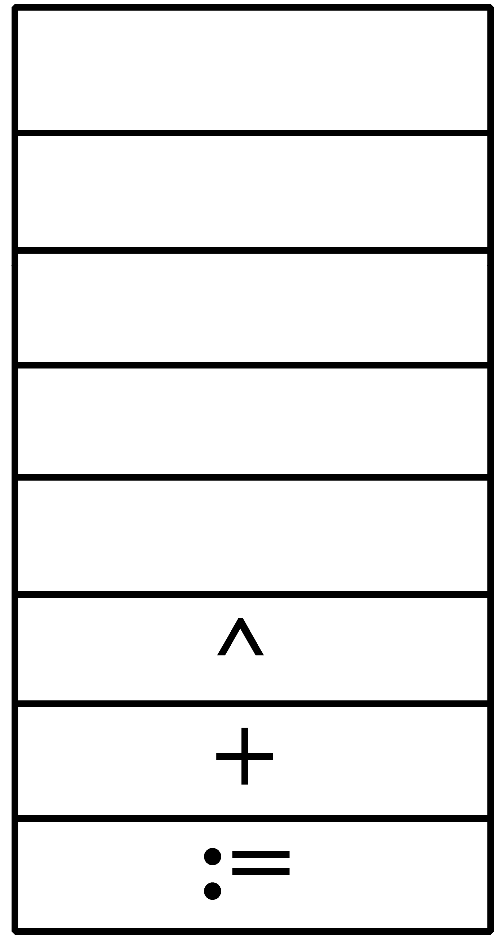


1. x:=a+b^3^2

Operátor jön, berakjuk a verembe a szabályoknak megfelelően.

Kimenet:

x a b

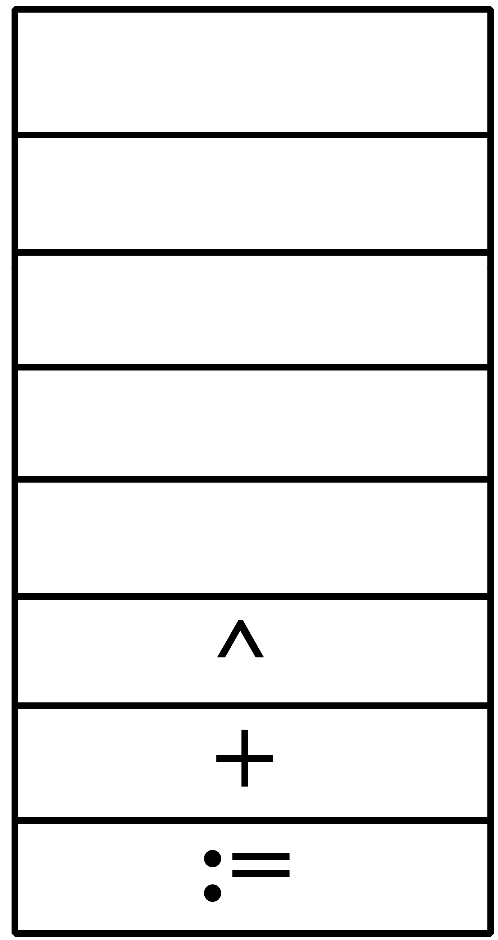


1. x:=a+b^3^2

Operandus jön, tehát kiírjuk a kimenetre.

Kimenet:

x a b 3

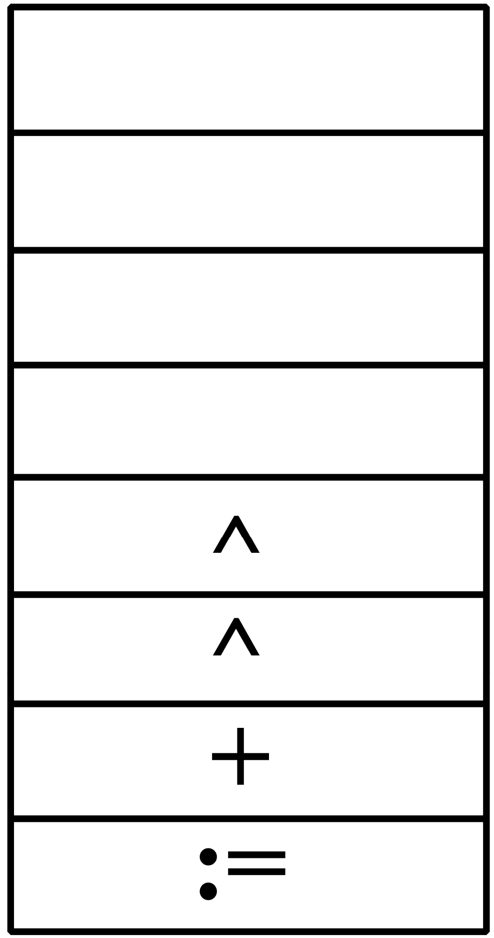


1. x:=a+b^3^2

Operátor jön, berakjuk a verembe a szabályoknak megfelelően.

Kimenet:

x a b 3

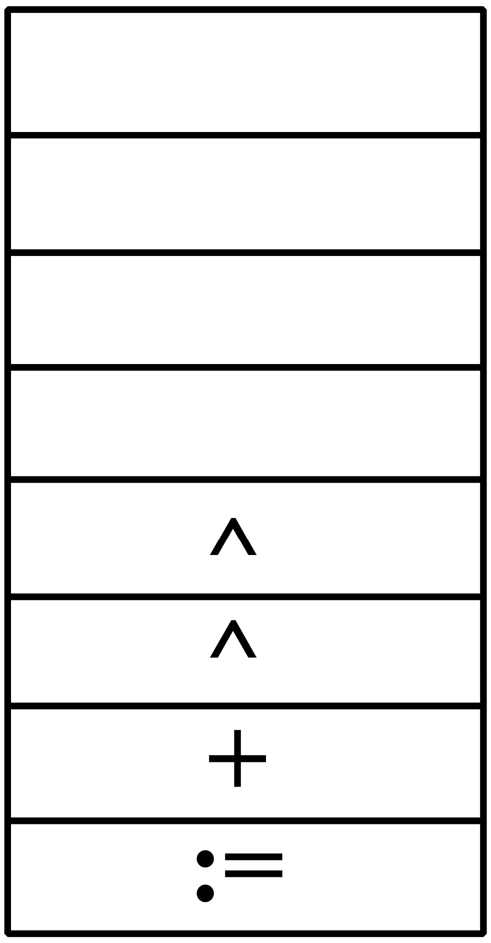


1. x:=a+b^3^2

Operandus jön, tehát kiírjuk a kimenetre.

Kimenet:

x a b 3 2



1. x:=a+b^3^2

A kifejezés „elfogyott”, így kiírjuk a verem tartalmát a kimenetre.

Kimenet:

x a b 3 2 ^ ^ + :=

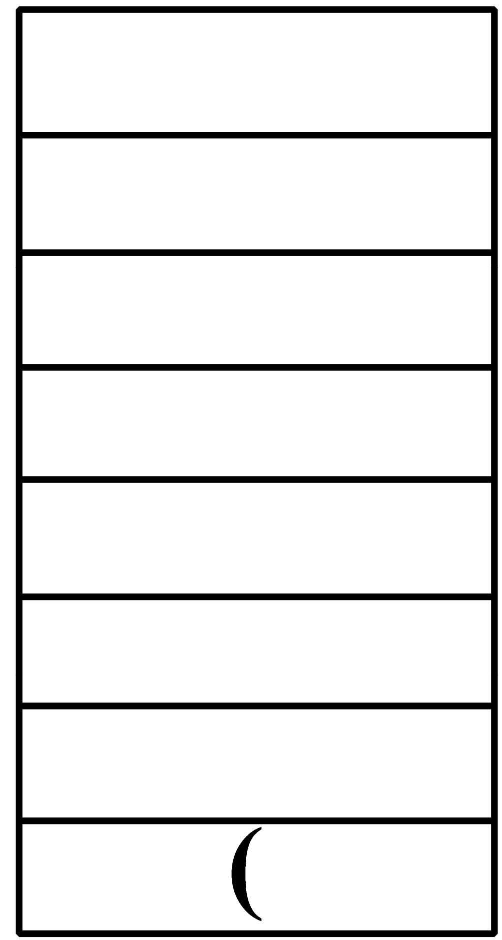
A verem kiürült.

2. (a+b)*(c-d)

Hozzuk lengyelformára a fenti kifejezést.

Nyitó zárójel az első jel, tehát betesszük a verembe.

A verem tartalma:

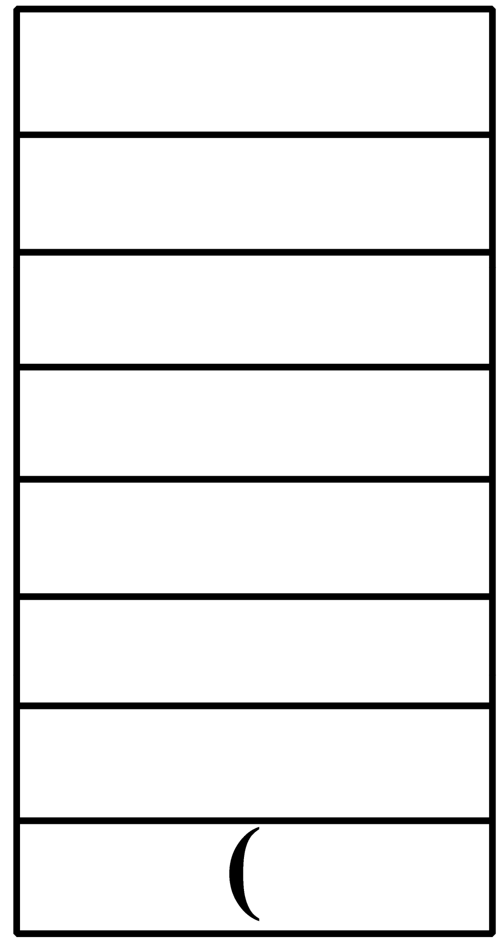


2. (a+b)*(c-d)

Operandus következik, tehát kiírjuk a kimenetre.

Kimenet:

a

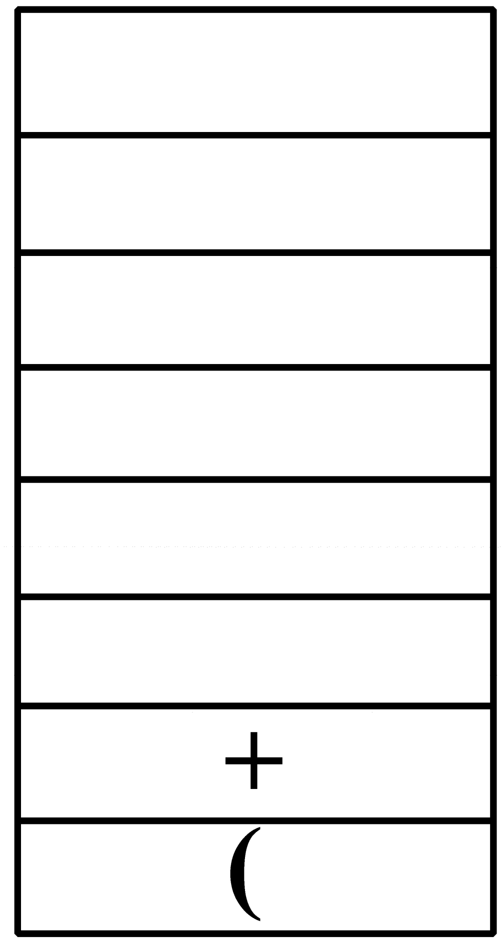


2. (a+b)*(c-d)

Operátor következik, tehát betesszük a verembe.

Kimenet:

a

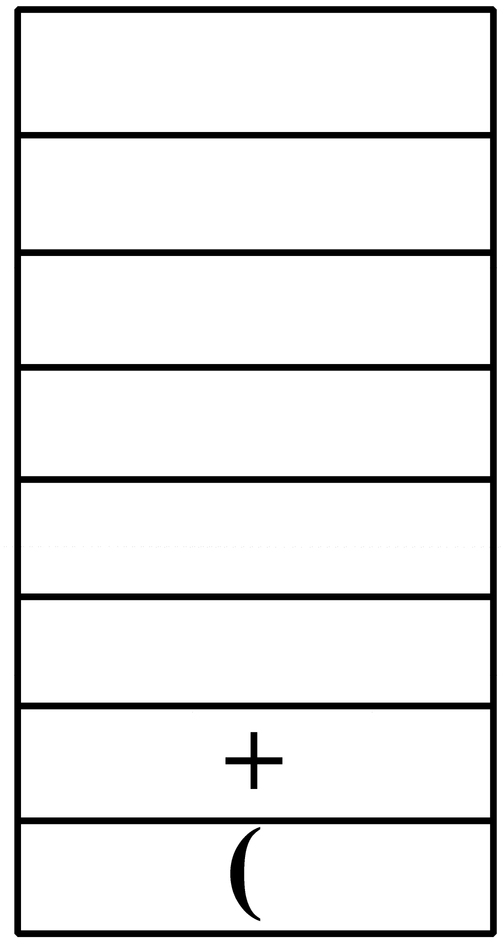


2. (a+b)*(c-d)

Operandus következik, tehát kiírjuk a kimenetre.

Kimenet:

a b



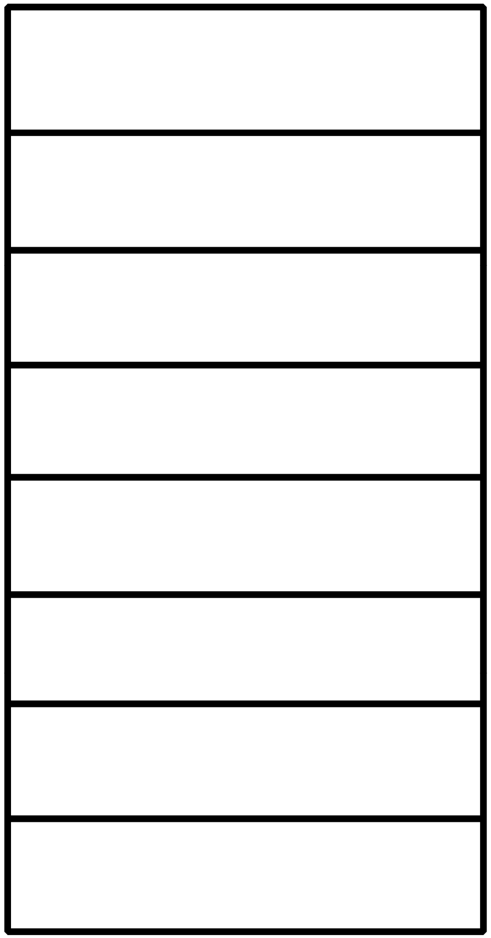
2. (a+b)*(c-d)

Csukó zárójel következik, tehát a nyitózárójelig mindent kiírunk a veremből, a zárójelpárt eldobjuk.

A verem újra üres lesz.

Kimenet:

a b +

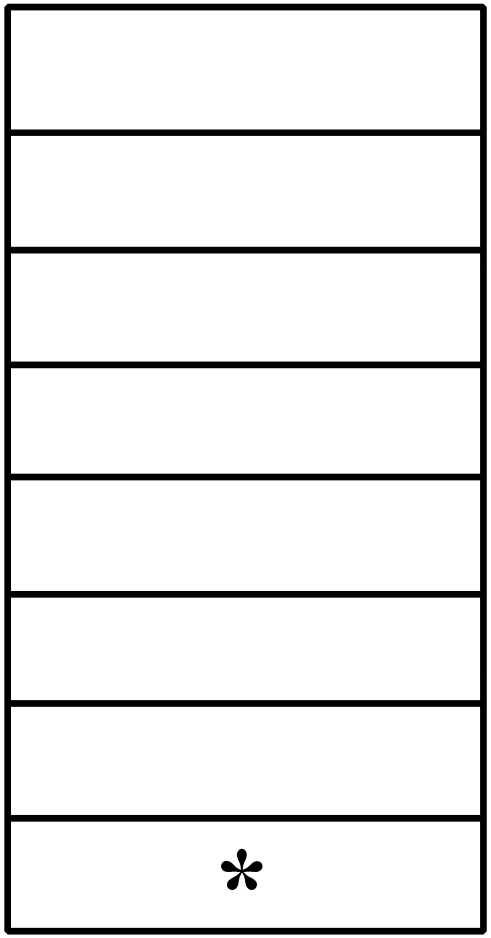


2. (a+b)*(c-d)

Operátor következik, tehát betesszük a verembe.

Kimenet:

a b +

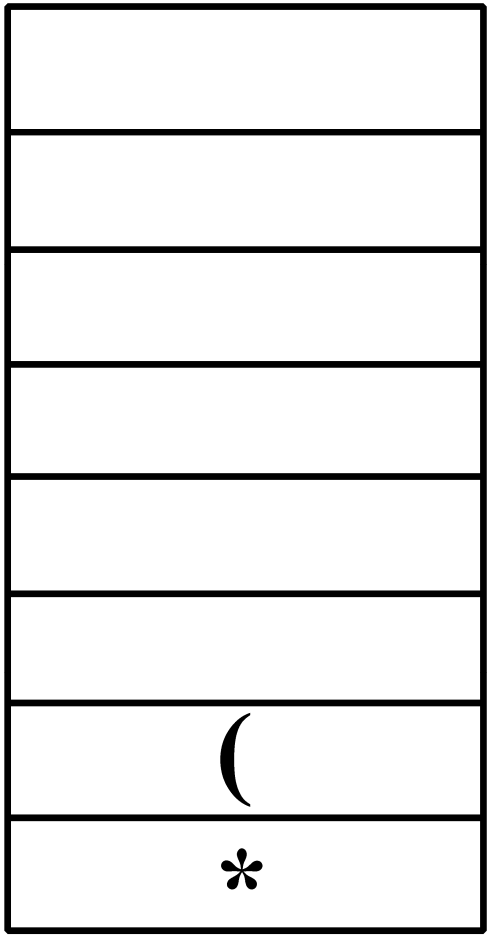


2. (a+b)*(c-d)

Nyitó zárójel következik, tehát betesszük a verembe.

Kimenet:

a b +

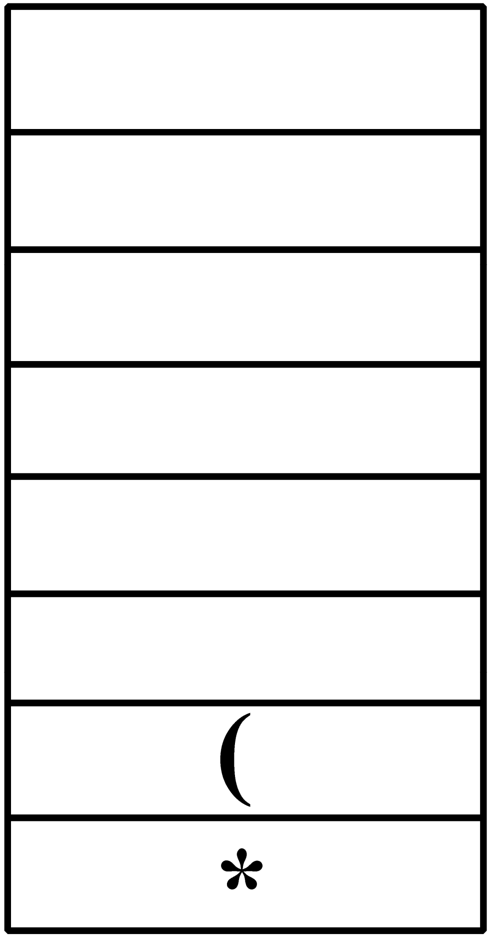


2. (a+b)*(c-d)

Operandus következik, tehát kiírjuk a kimenetre.

Kimenet:

a b + c

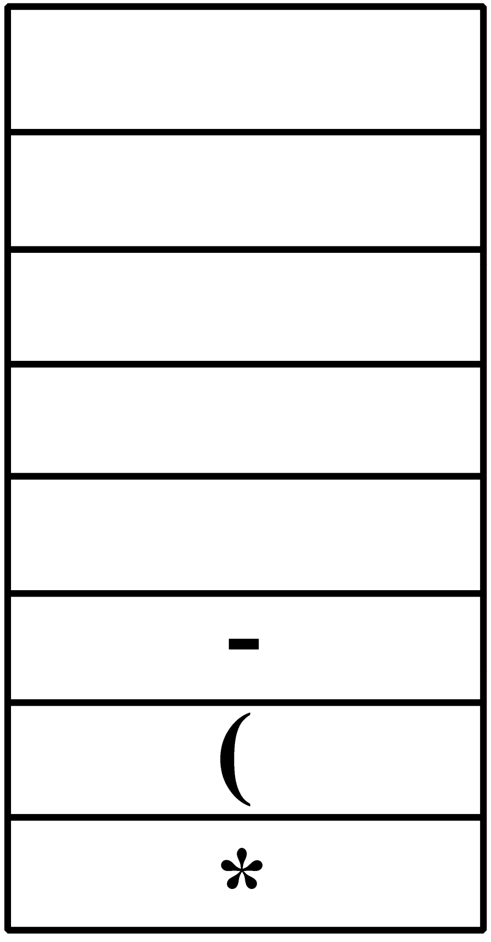


2. (a+b)*(c-d)

Operátor következik, tehát betesszük a verembe.

Kimenet:

a b + c

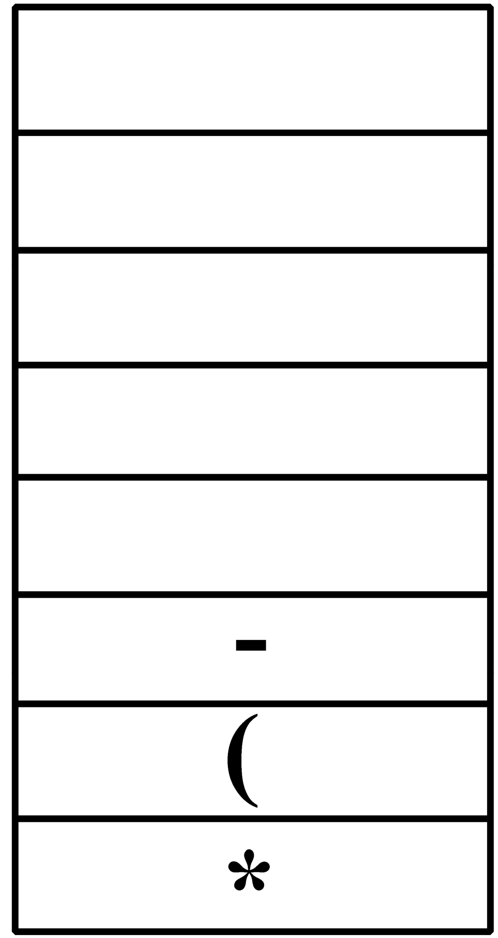


2. (a+b)*(c-d)

Operandus következik, tehát kiírjuk a kimenetre.

Kimenet:

a b + c d

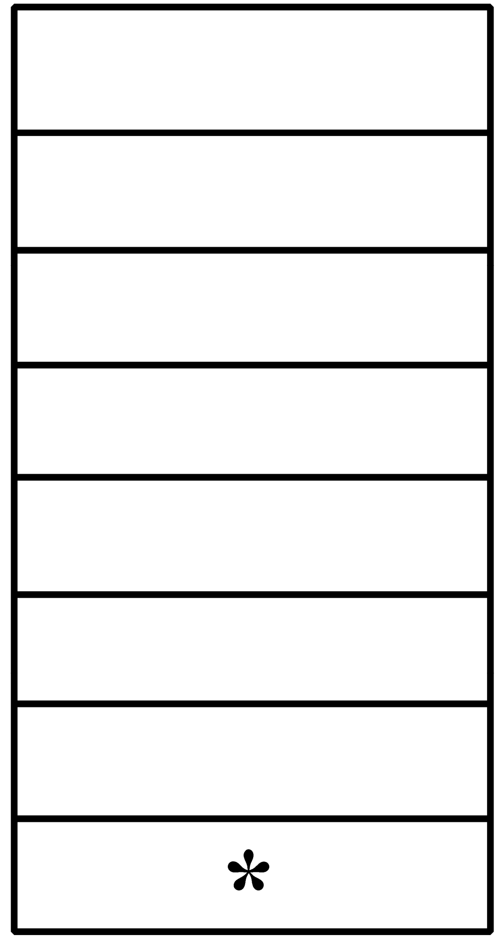


2. (a+b)*(c-d)

Csukó zárójel következik, tehát a nyitózárójelig mindent kiírunk a veremből, a zárójelpárt eldobjuk.

Kimenet:

a b + c d -

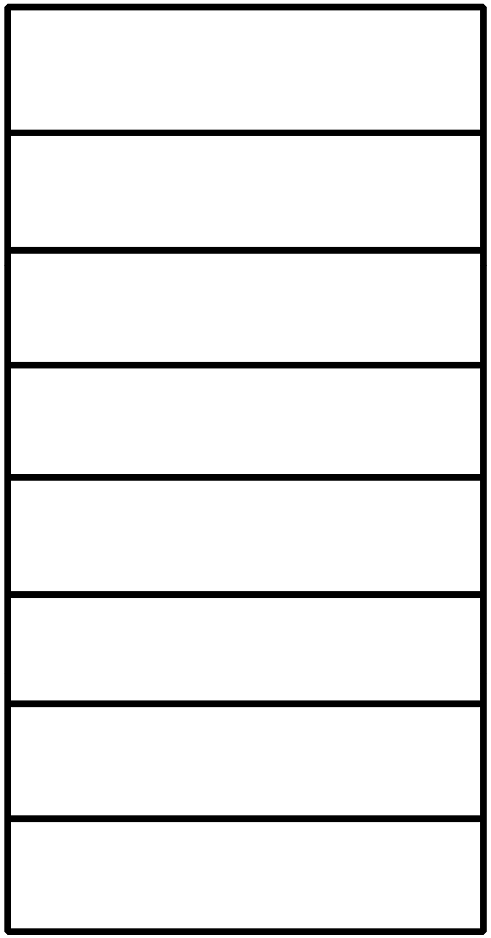


2. (a+b)*(c-d)

Végül ürítsük ki a vermet, mivel a kifejezés végére értünk.

Kimenet:

a b + c d - *



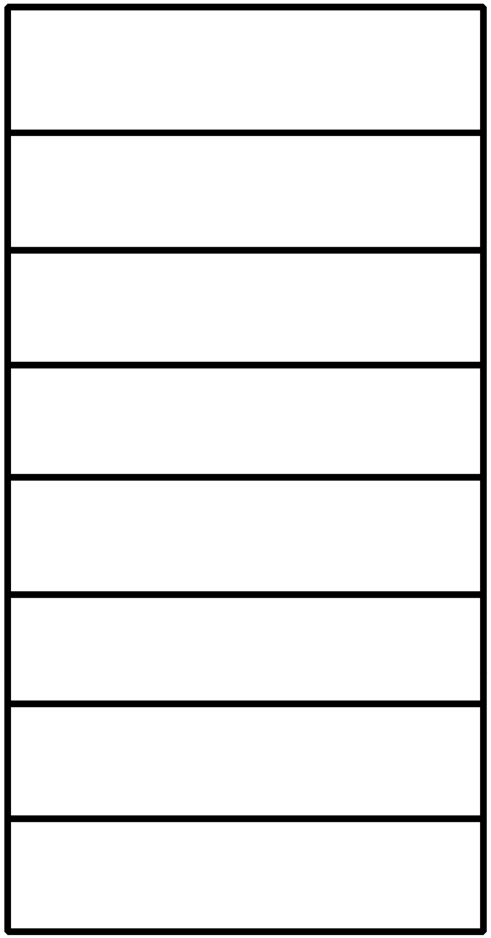
3. x:=(a+b)*c-(a+b)^2

Hozzuk lengyelformára a fenti kifejezést.

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x

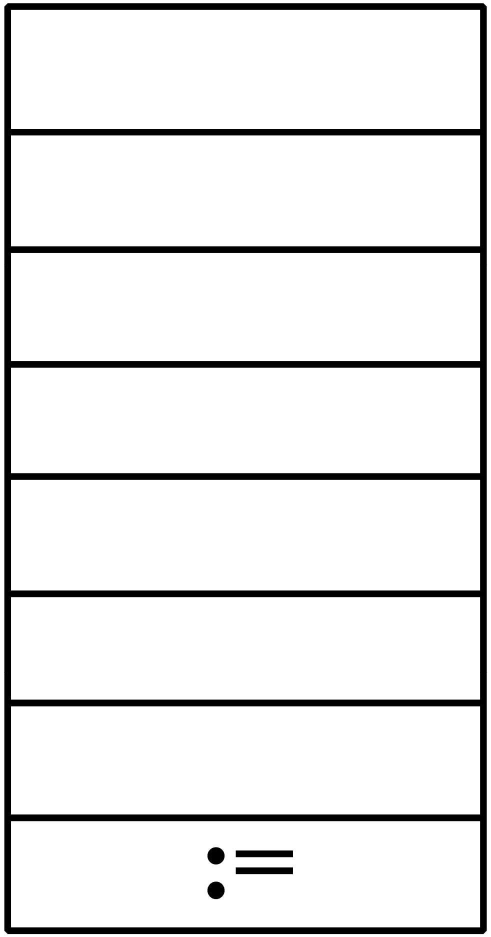


3. x:=(a+b)*c-(a+b)^2

Operátor következik, így betesszük a verembe.

Kimenet:

x

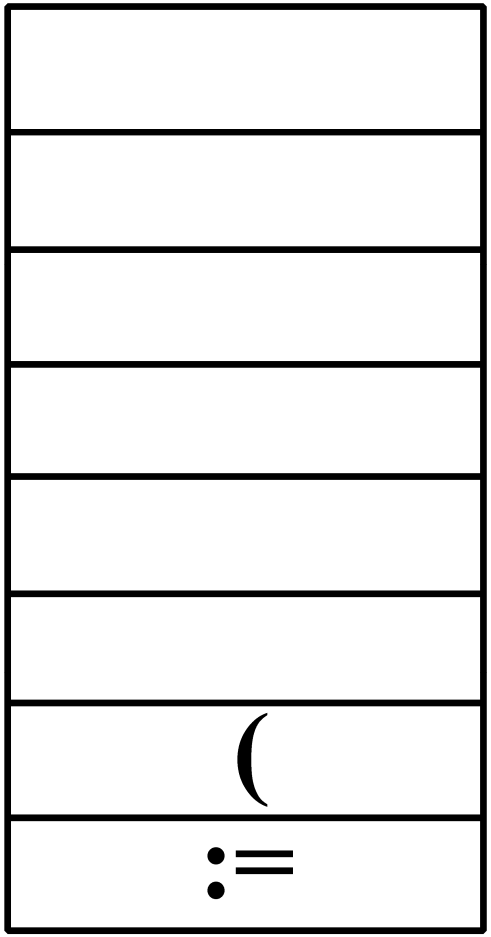


3. x:=(a+b)*c-(a+b)^2

Nyitó zárójel következik, így betesszük a verembe.

Kimenet:

x

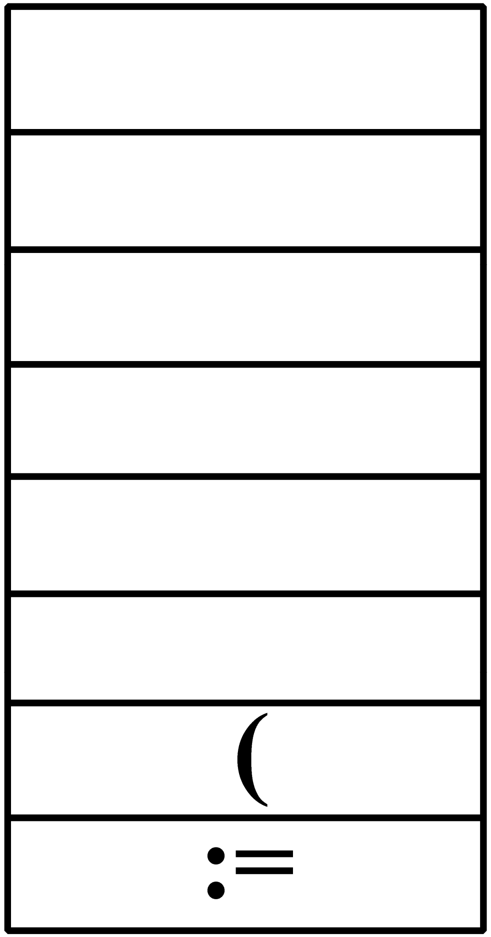


3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a

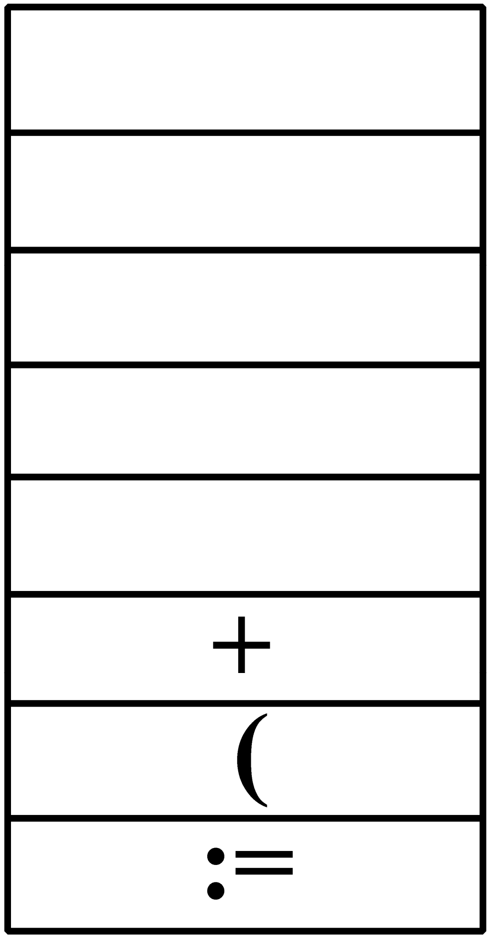


3. x:=(a+b)*c-(a+b)^2

Operátor következik, így betesszük a verembe.

Kimenet:

x a

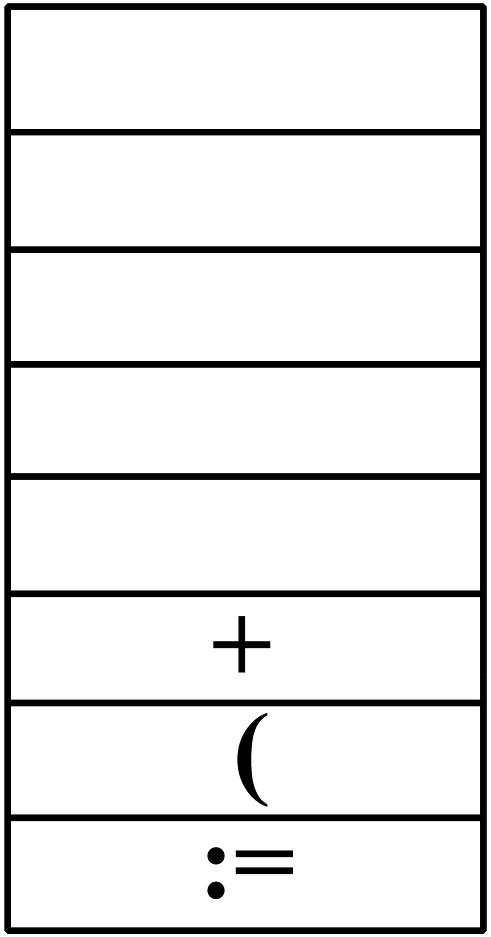


3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b

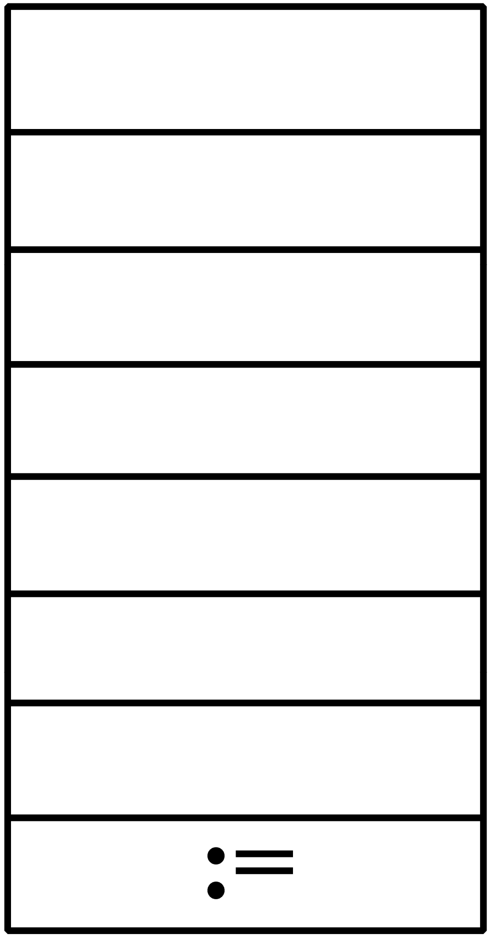


3. x:=(a+b)*c-(a+b)^2

Csukó zárójel következik, így a nyitó zárójelig mindent kiürítünk a veremből, majd a zárójel-párt eldobjuk.

Kimenet:

x a b +

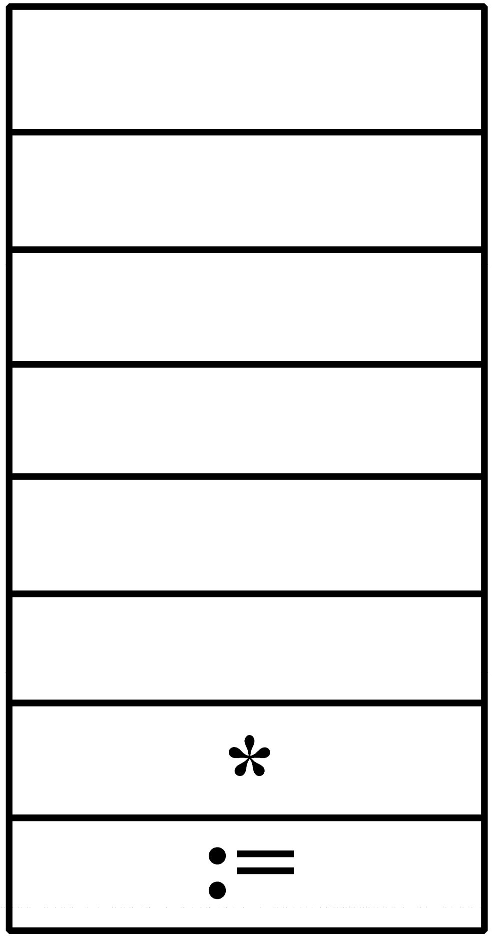


3. x:=(a+b)*c-(a+b)^2

Operátor következik, így betesszük a verembe.

Kimenet:

x a b +



3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

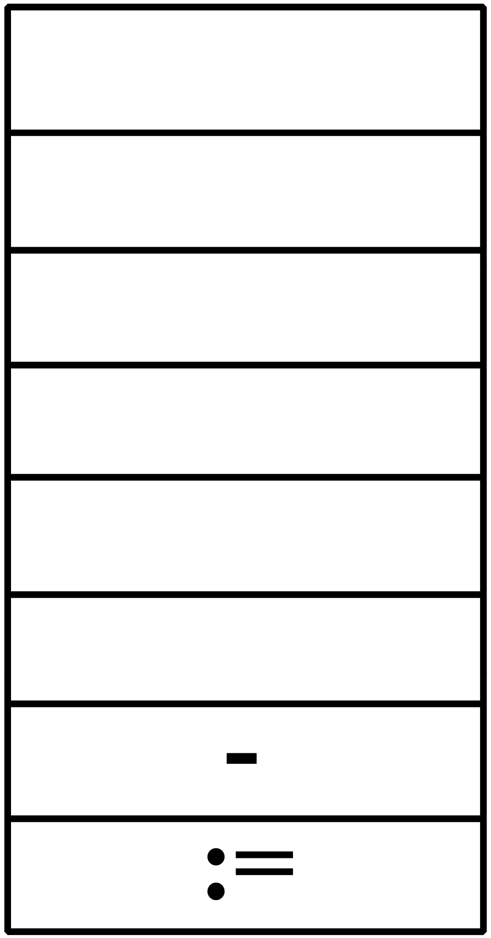
x a b + c

3. x:=(a+b)*c-(a+b)^2

Operátor következik, mivel nála magasabb precedenciájú operátor van a veremben, azt kiírjuk, a „-”-t pedig betesszük a verembe.

Kimenet:

x a b + c *

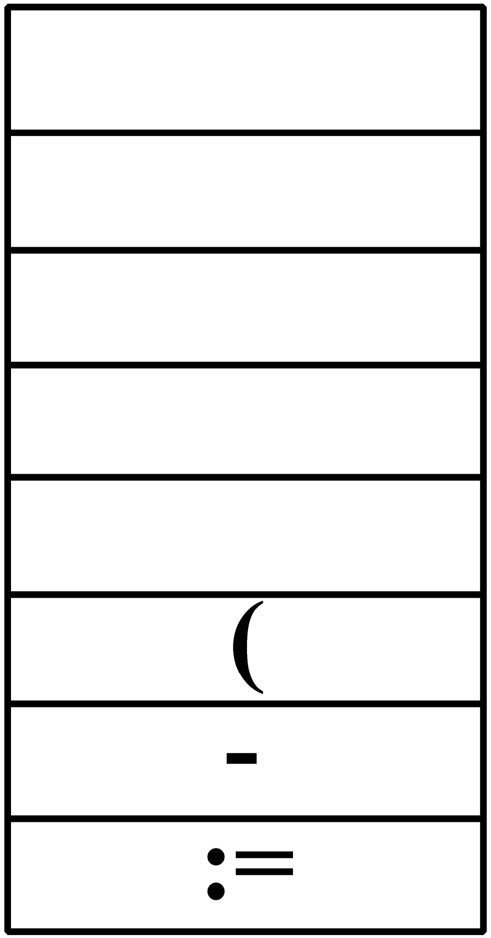


3. x:=(a+b)*c-(a+b)^2

Nyitó zárójel következik, így betesszük a verembe.

Kimenet:

x a b + c *



3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b + c * a

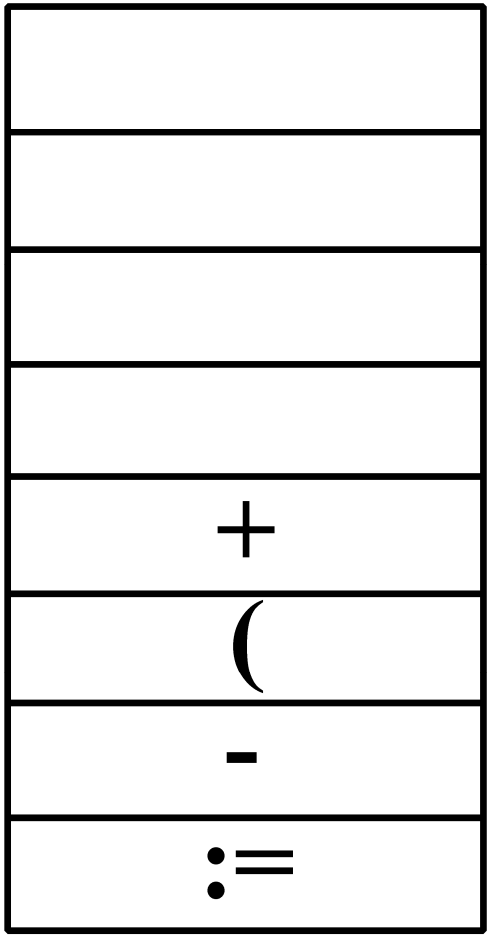
(
-
:=

3. x:=(a+b)*c-(a+b)^2

Operátor következik, így betesszük a verembe.

Kimenet:

x a b + c * a



3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b + c * a b

+
(
-
:=

3. x:=(a+b)*c-(a+b)^2

Csukó zárójel következik, így a nyitó zárójelig mindent kiürítünk a veremből, majd a zárójel-párt eldobjuk.

Kimenet:

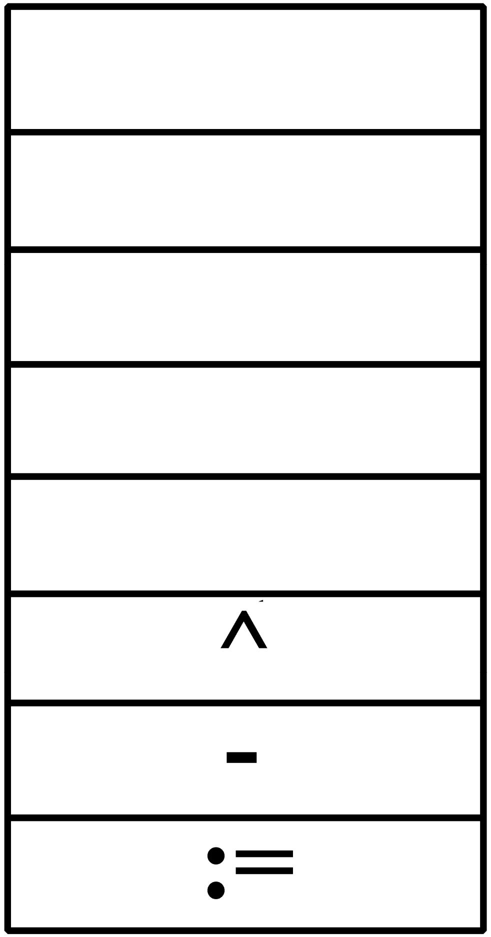
x a b + c * a b +

3. x:=(a+b)*c-(a+b)^2

Operátor következik, így betesszük a verembe.

Kimenet:

x a b + c * a b +

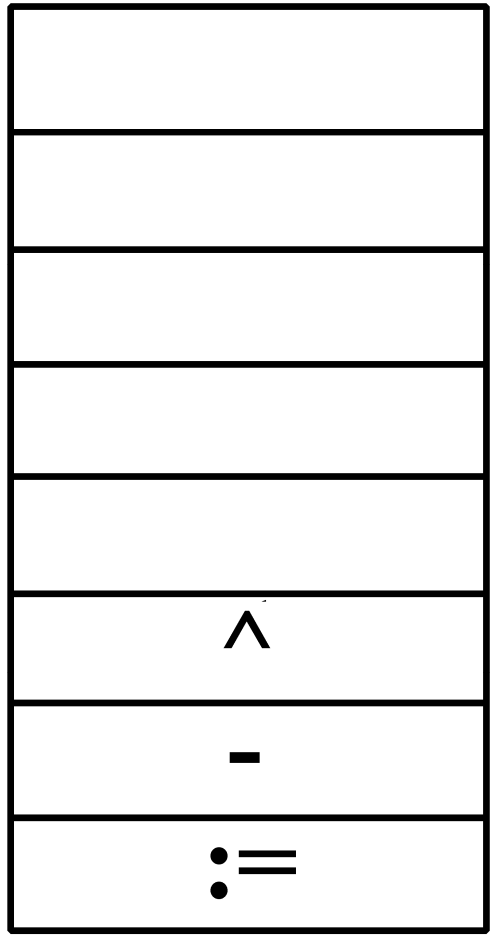


3. x:=(a+b)*c-(a+b)^2

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b + c * a b + 2

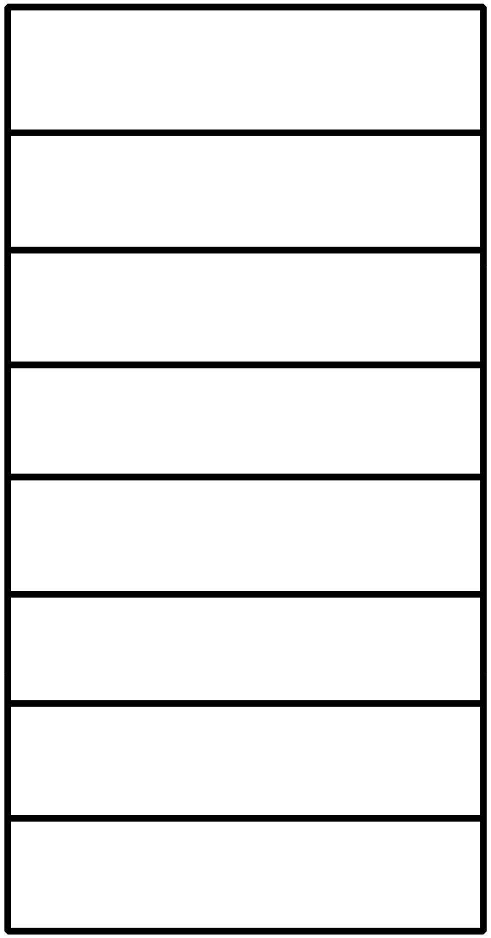


3. x:=(a+b)*c-(a+b)^2

A kifejezés végére értünk, így kiürítjük a vermet.

Kimenet:

x a b + c * a b + 2 ^ - :=



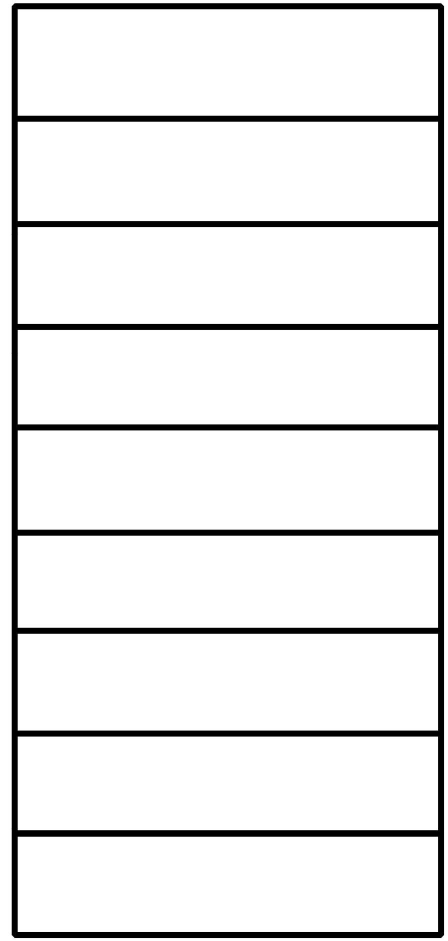
4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Hozzuk lengyelformára a fenti kifejezést.

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x

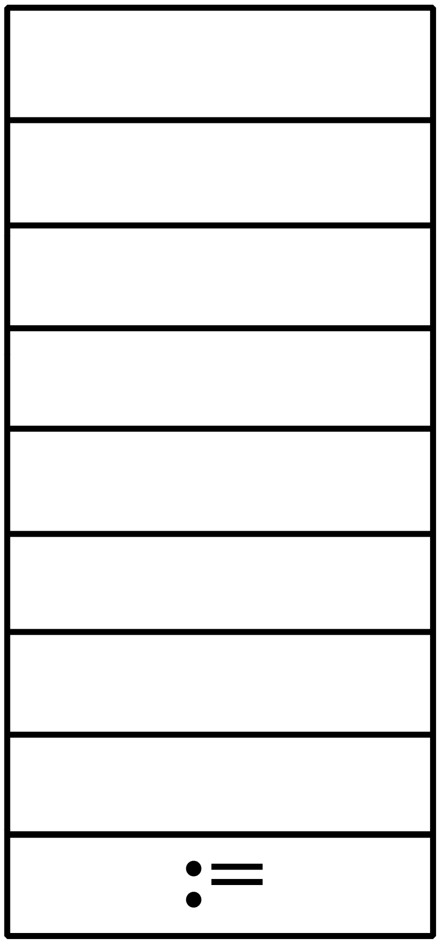


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, így betesszük a verembe.

Kimenet:

x

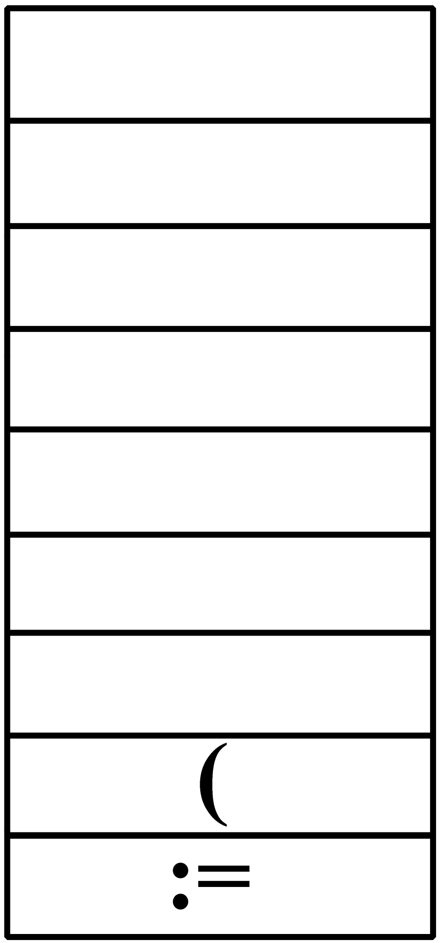


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Nyitó zárójel következik, így betesszük a verembe.

Kimenet:

x

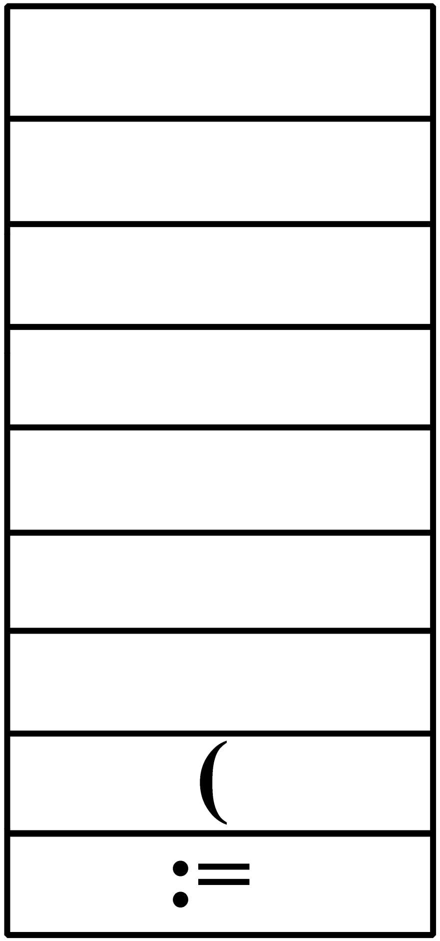


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operatór következik, így betesszük a verembe.

Kimenet:

x a

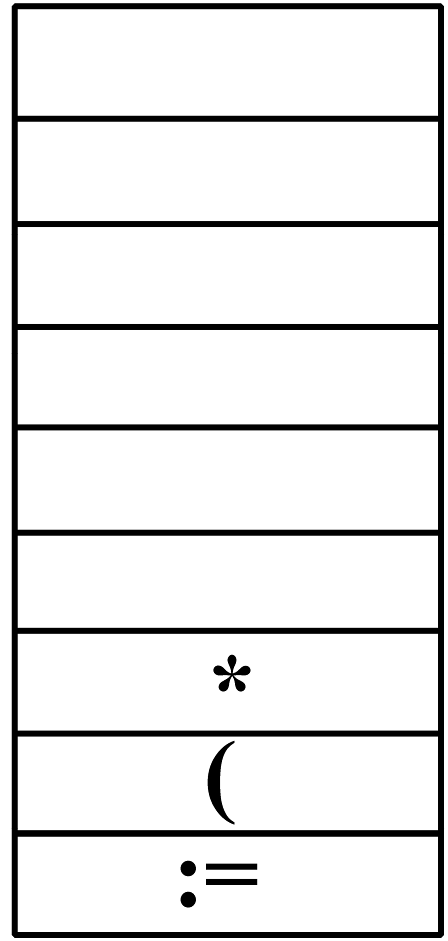
*
(
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b

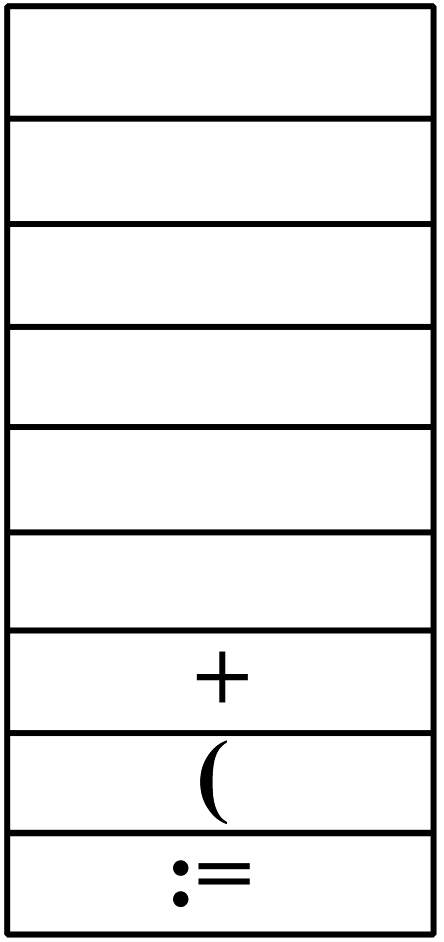


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, a nála magasabb precedenciájú operátort kiírjuk, a „+”-t betesszük a verembe.

Kimenet:

x a b *

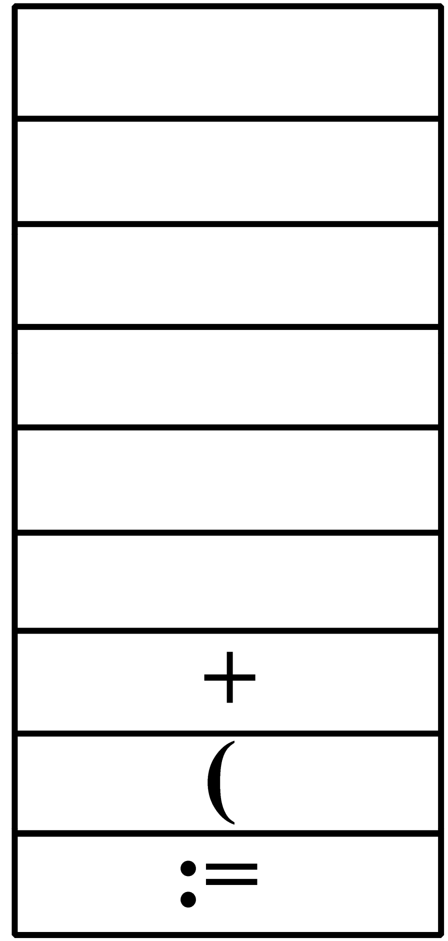


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

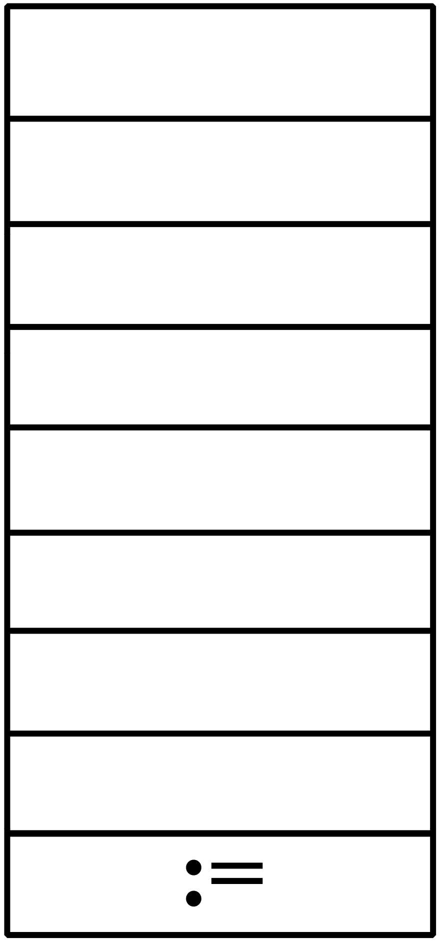
Kimenet:

x a b * 1



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Csukó zárójel következik, így kiírjuk a kimenetre a vermet a nyitó zárójelig, a zárójelpárt eldobjuk.



Kimenet:

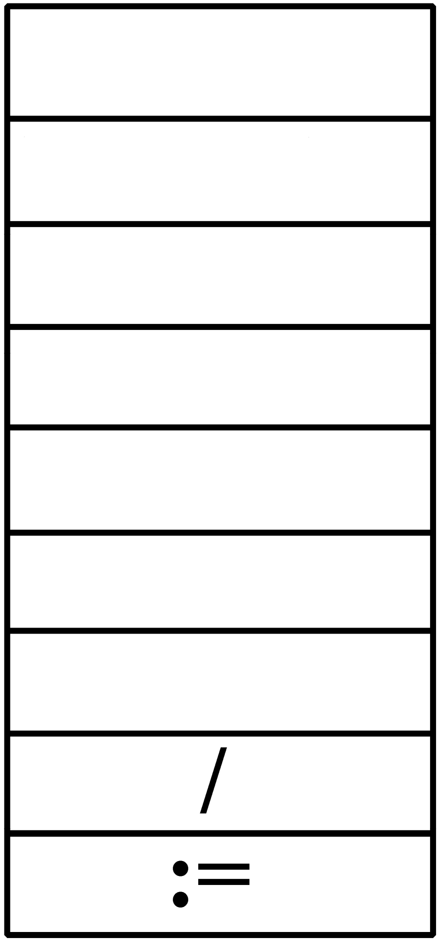
x a b * 1 +

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operatór következik, így betesszük a verembe.

Kimenet:

x a b * 1 +



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

2 db nyitó zárójel következik, így betesszük azokat a verembe.

Kimenet:

x a b * 1 +

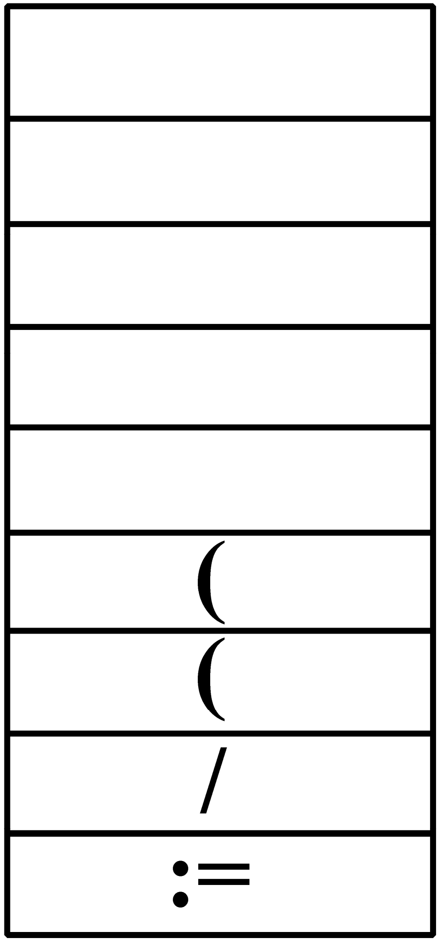
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x

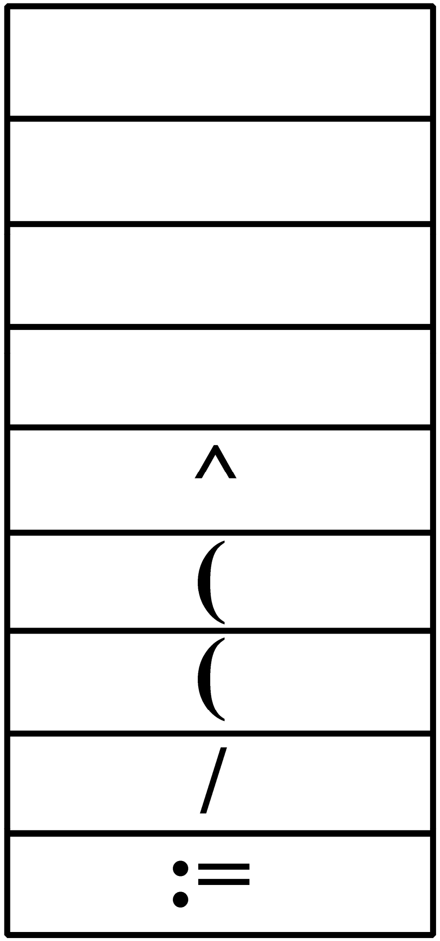


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, így betesszük a verembe.

Kimenet:

x a b * 1 + x



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y

^
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, és mivel jobbról-balra szabály van, a vele azonos precedenciájú operátort nem írjuk ki. Az operátort betesszük a verembe.

Kimenet:

x a b * 1 + x y

^
^
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2

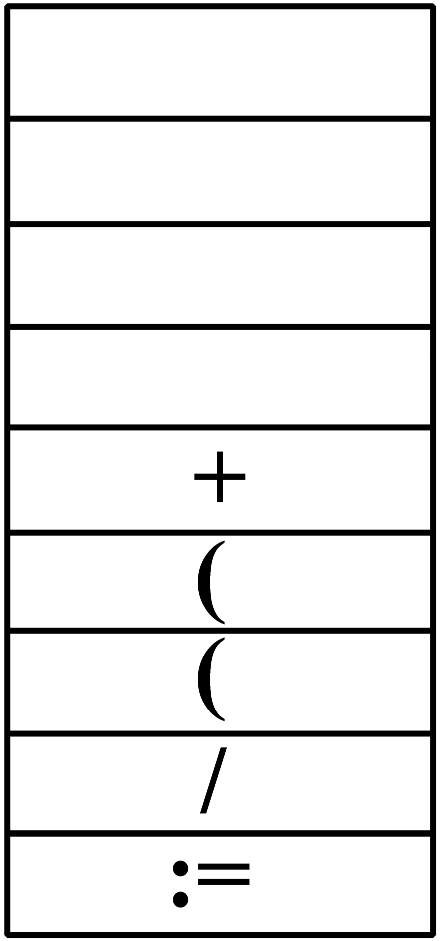
^
^
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operatór következik, 2 db nála magasabb precedenciájú operátor van a veremben így azokat kiírjuk a kimenetre. A „+” operátort a verembe rakjuk.

Kimenet:

x a b * 1 + x y 2 ^ ^



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2 ^ ^ u

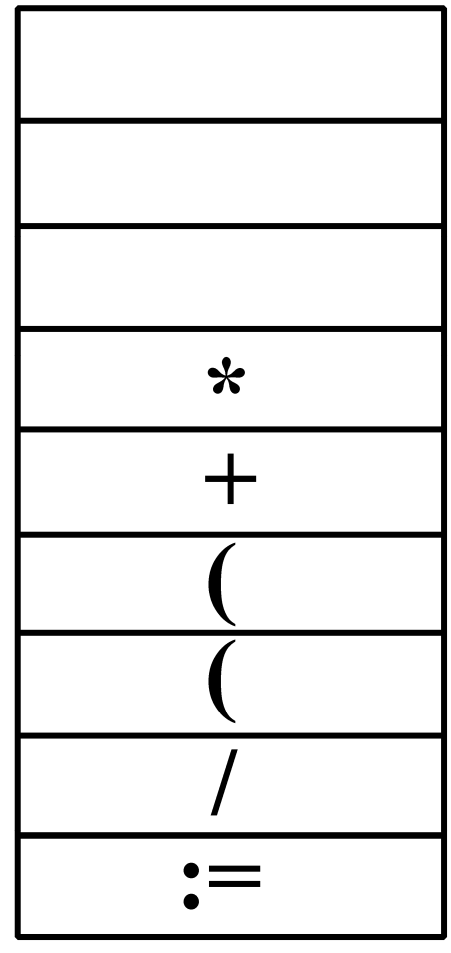
+
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, nincs vele egyenlő, illetve nála magasabb precedenciájú operátor a veremben, így betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u

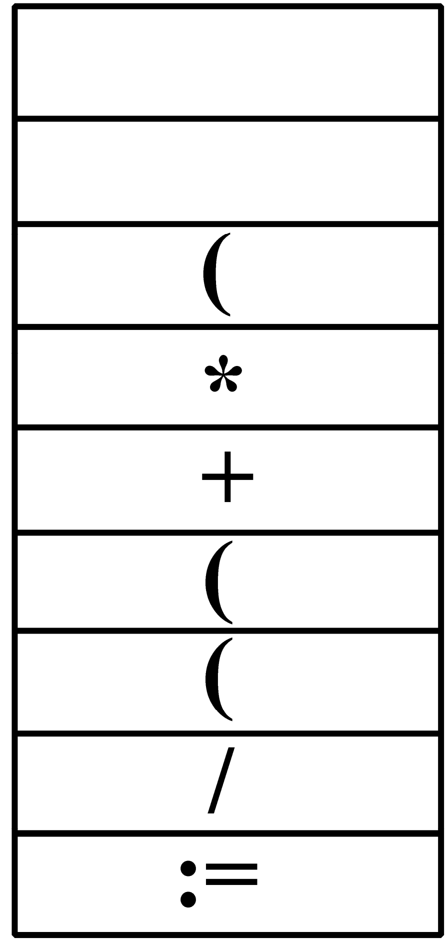


4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Nyitó zárójel következik, így betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v

(
*
+
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, így
betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v

-
(
*
+
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3

-
(
*
+
(
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Csukó zárójel következik, így kiírjuk a kimenetre a következő nyitó zárójelig az összes operátort, majd a zárójelpárt eldobjuk.

*
+
(
(
/
:=

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 -

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Szintén csukó zárójel következik, így kiírjuk a kimenetre a következő nyitó zárójelig az összes operátort, majd a zárójelpárt eldobjuk.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * +

(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, így betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * +

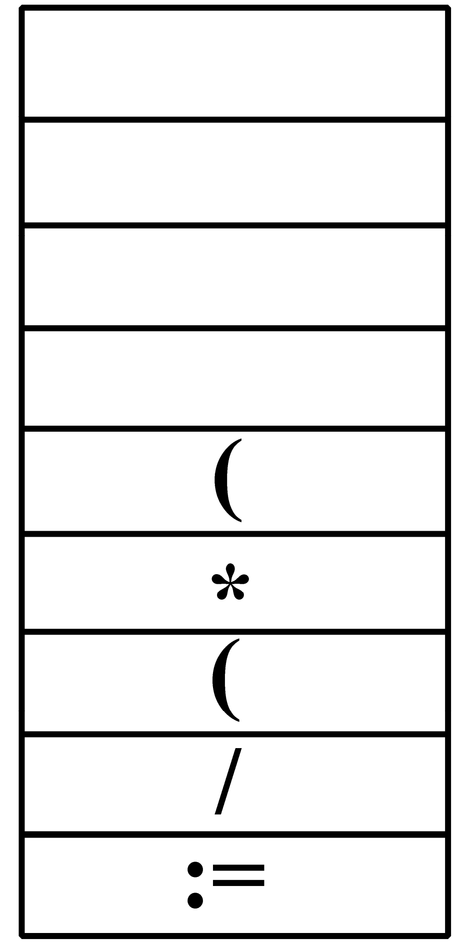
*
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Nyitó zárójel következik, így betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * +



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f

(
*
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, így betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f

+
(
*
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f g

+
(
*
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operátor következik, mivel vele azonos precedenciájú operátor van a veremben, azt kiírjuk, a „-”-t pedig betesszük a verembe.

Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f g +

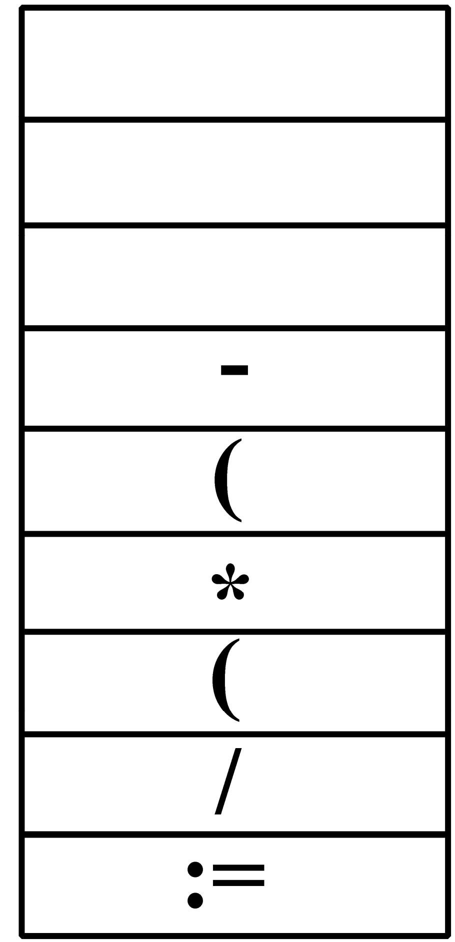
-
(
*
(
/
:=

4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

Operandus következik, így kiírjuk a kimenetre.

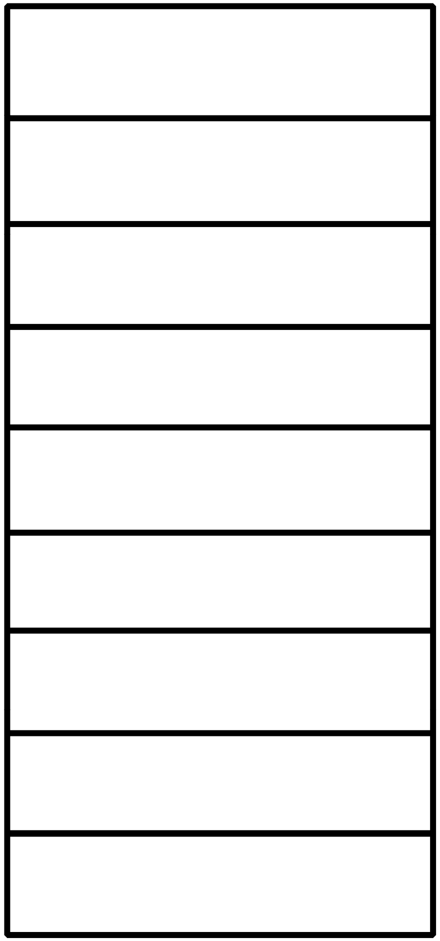
Kimenet:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f g + 4



4. x:=(a*b+1)/((x^y^2+u*(v-3))*(f+g-4))

2 db csukó zárójel következik, azután a kifejezésnek vége lesz, így lényegében mindent kiírunk a veremből, természetesen a zárójeleket eldobjuk.



A végeredmény:

x a b * 1 + x y 2 ^ ^ u v 3 - * + f g + 4 - * / :=