

"Wolfram Mathematica" īsa lietošanas instrukcija

Dr.Sc.Ing. Jānis Voitkāns

1. Simboli

1.1. Simbols

Tam nav skaitliskas vērtības. Tas tiek uzrādīts zilā krāsā.

Piemērs

a , b_1 , I_2 , R_3

1.2. Piešķiršanas zīme = (ir)

Veic labās puses (rhs) skaitliskās vērtības vai simbola piešķiršanu kreisajai pusei (lhs)

Tā nav vienādības zīme, kā pierasts.

Piemērs

$a = 3$

rhs vērtība 3 tiek piešķirta lhs simbolam a

Pēc darbības izpildes simbols a iekrāsojas melns. Tas nozīmē, ka Wolframs simbolu a atpazīst kā skaitli

$a = b$

rhs vērtība b tiek piešķirta lhs simbolam a

Pēc a krāsas var noteikt, vai tas ir ieguvis skaitlisku vērtību, vai saglabājis simbola statusu

1.3. Vienādības zīme == (dubults ir)

Tā ir zīme, kas parāda vienādību, bet neveic piešķiršanu.

Piemērs

$a == b$

Dotās vienādības labā un kreisā puse saglabā to statusu, kas tām jau ir

1.4. Ir aizņemtie simboli, kurus nedrīkst lietot savādāk, kā tikai tam, kam tie domāti.

Piemērs

I , i - *imaginārā vienība*

π , Pi - *skaitlis $\pi \approx 3.14 \dots$*

∞ - *bezgalība*

$^\circ$ - *grādi, lieto, ja leņķis jāparāda grādos, nevis radiānos*

e , E - *eksponentālā konstante (arī naturālā logaritma bāze) $e \approx 2.72 \dots$*

C - *konstante, kas tiek lietota dažādos analītiskos aprēķinos (piemēram – integrēšanas konstante)*

1.5. Zīme // , kas parāda,
ka komanda darbojas pa kreisi no tās (lhs // komanda).

Piemērs

`30 ° // Sin`

atbildē dod 0.5

`$\frac{1}{3}$ // N`

atbildē dod 0.333333

`lhs // Simplify`

atbildē dod vienkāršotu kreisās puses izteiksmi

1.6. Aizvietošanas zīme /. ,

kas veic rhs skaitļa vai simbolapieskiršanu attiecīgam kreisās
puses simbolam (lhs /. rhs).

Piemērs

`a b /. a → 2`

atbildē dod 2 a

`a b + c /. {a → 2, c → d}`

atbildē dod 2 b + d

Ar šīs zīmes palīdzību var veikt doto skaitlisko vērtību ievietošanu
analītiskā atrisinājumā, lai iegūtu skaitlisko rezultātu
Bultiņu raksta ar `- >` zīmju palīdzību, vienu pēc otras

1.7. Iekavas

1.7.a. Parastās iekavas () uzrāda matemātisko darbību prioritāti.

Piemērs

`(a + b) c`

summēšana tiek veikta vispirms, pēc tam reizināšana

1.7.b. Kvadrātiskās iekavas [],

tajās ievieto funkciju argumentus un objektus,
uz kuriem iedarbojas komandas.

Piemērs

`Sin[x]`

`Solve[{■ == □, □ == □}, {□, □}]`

1.7.c. Figūriekavas { },

lieto jebkuru skaitļu vai simbolu masīvu uzrādīšanai.

Piemērs

$$\frac{\{2, 3, a, 5, b, 3\}}{2}$$

atbildē dod $\left\{1, \frac{3}{2}, \frac{a}{2}, \frac{5}{2}, \frac{b}{2}, \frac{3}{2}\right\}$

1.8. Simbolu iekrāsojums

1.8.a. Ja iekrāsojums ir melns, tad tam ir piešķirta skaitliska vērtība.

Piemērs

1, 2, I, C *utt.*

1.8.b. Ja iekrāsojums ir zils, tad simbolam nav skaitliskās vērtības, tas ir tīrs simbols.

Piemērs

R1, a, I2 *utt.*

1.8.c. Ja iekrāsojums ir zaļš, tad tas ir vienādojuma (vai vienādojumu sistēmas) nezināmais.

Piemērs

Solve[a x + b == 0, x] nezināmais x ir zaļā krāsā

2. Programma

2.1. Programma sastāv no komandām, kas atdalītas ar "Enter".

2.2. Programmas garumu parāda iekava darba lapas labajā pusē, kas ar katru Enter stiepjas garāka.

2.3. Programmas izpilde tiek ierosināta ar (Shift + Enter), vai arī ar tālo "Enter".

2.4. Katras komandas rezultāts tiek uzrādīts uzreiz pēc programmas iekavas beigām gaišākā šriftā.

2.5. Komandu rezultātus, kurus mums nav nepieciešams redzēt, vajag noslēpt, programmā pēc attiecīgās komandas ieliekot semikolu (;).

2.6. Programmas izpilde tiek saukta par "Evolution".

2.7. Ja programmas izpildes laiks ir par garu, tad jāuzklikšķina uz "Abort Evolution" sadaļā "Evolution".

2.8. Ja nepieciešams programmu atgriest sākuma stāvoklī un simboliem notīrīt piešķirtās skaitliskās vērtības, tad jānospiež "Quit Kernel" šajā pašā sadaļā "Evolution".

3. Help

3.1. Visu komandu aprsksti un paskaidrojumi ir Helpā, kuru var paņemt, attiecīgo komandu vai simbolu iekrāsojo, vai ieklišķinot tajā kursoru, un tad nospiežot taustiņu F1.

3.2. Helpā ir parādīts apraksts un piemēri, kas darbojas, tos var pārkopēt savā programmā un pielietot.

3.3. Piemēru skaitu var palielināt, klikšķinot uz trijstūrīšiem kreisajā pusē.

3.4. Helpā tiek parādīta arī īsa matemātikas teorija par attiecīgo tēmu.

4. Programmu lietošanas pamatnoteikumi.

4.1. Ja simbolam ir piešķirta skaitliskā vērtība, tad tā saglabājas arī citās programmās un darba lapās dotā Kernela ietvaros, to vairs nedrīkst lietot kā mainīgo komandās `Solve[^]` vai `Plot[]`.

4.2. Ja nepieciešams simbolu pārvērst atpakaļ par simbolu, tad jālieto notīrīšanas komandu `Clear[^]`, vai restartu "Quit Kernel".

4.3. Ja komanda dod rezultātu formā ar bultiņu \rightarrow , piemēram `{I1 \rightarrow 0.5, I2 \rightarrow 0.8}`, tad vērtības 0.5 un 0.8 vēl nav piešķirtas simboliem I1 un I2, lai šīs vērtības pielietotu tālāk, ir jāveic to aktivizācija.

4.4. Jāpiešķir vienādojumu sistēmu kādam simbolam (piem. I_k), tad ar to aktivizē attiecīgā atrisinājuma simbolu (piem. I_{k1} , I_{k2} , I_{k3}).

$I_k = \text{Solve}[\{\square = \square, \square = \square, \square = \square\}, \{I_1, I_2, I_3\}]$

Iegūst, piemēram, $\{I_1 \rightarrow 0.5, I_2 \rightarrow 0.8, I_3 \rightarrow 1.2\}$, to jāaktivizē

$I_{k1} = I_1 /. I_k[[1]]$

$I_{k2} = I_2 /. I_k[[1]]$

$I_{k3} = I_3 /. I_k[[1]]$

Rezultātā I_{k1} , I_{k2} , I_{k3} iegūst vērtības 0.5, 0.8 un 1.2.

4.5. Ja vajag aktivizēt atrisinājumu, kas nav nosaukti, masīvu, piemēram $| I_k |$,

ja zināma pretestību matrica $| R_m |$ un EDS matrica $| E_m |$

$| I_k | = | R_m |^{-1} | E_m |$

Rezultātā iegūst $\{0.5, 0.8, 1.2\}$, to aktivizē, zinot konkrēto atrisinājuma numuru pēc kārtas

$I_{k1} /. I_k[[1]]$

$I_{k2} /. I_k[[2]]$

$I_{k3} /. I_k[[3]]$

Tālāk kontūrstrāvas I_{k1} , I_{k2} un I_{k3} var brīvi izmantot tālākos aprēķinos.

4.6. Programmā skaitliskās vērtības ir jāizmanto kā simboli, tās nedrīkst pārkopēt kā skaitļus.

4.7. Kā simboli ir jāizmanto skaitļi π un e , nedrīkst lietot 3.14 un 2.72.

4.8. Vienkāršu komandu (saskaitīšana, reizināšana, kāpināšana, utt.) vietā nedrīkst lietot sarežģītās komandas, kas dod to pašu rezultātu, jo to izpildes laiks ir daudz lielāks. Piemēram

$I1 = I_{k1}$

$I2 = I_{k1} + I_{k2}$

$I3 = I_{k3}$

nedrīkst izpildīt kā

$\text{Solve}[\{I1 == I_{k1}, I2 == I_{k1} + I_{k2}, I3 == I_{k3}\}, \{I1, I2, I3\}]$,

kaut arī rezultāts ir sakrītošs. Bez tam ar Solve iegūtās vērtības vēl ir jāaktivizē.

4.9. Ja komandu vajag izpildīt, bet rezultātu nevajag parādīt,
tad aiz komandas jāliek semikolu

Piemēram

$R1 = 2\theta;$

$I2 = Ik1 + Ik2;$