

EDUCATIONAL TOOL FOR FIRST ORDER LOGIC

Alexandra Nyitraiova

ČO ROBÍM?

-asistent pre dokazovanie tablovým kalkuľom

-nedokazuje za mňa

-validuje dôkaz a vyznačuje chyby

-dôkaz vizualizuje ako stromovú štruktúru,

$((\neg p \rightarrow \neg r) \leftrightarrow (r \rightarrow p))$ je tautológia

(1)	F $((\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p)) \wedge ((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r))$		[1]	X						
(2)	F $(\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p)$	[1]	X	(11)	F $((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r))$	[1]	X			
(3)	T $(\neg p \rightarrow \neg r)$	[2]	X	(12)	T $(r \rightarrow p)$	[11]	X			
(4)	F $(r \rightarrow p)$	[2]	X	(13)	F $(\neg p \rightarrow \neg r)$	[11]	X			
(5)	T r	[4]	X	(14)	T $\neg p$	[13]	X			
(6)	F p	[4]	X	(15)	F $\neg r$	[13]	X			
(7)	F p	[3]	X	(16)	F p	[14]	X			
(8)	T p	[7]	X	(17)	T r	[15]	X			
	* 8 6	X	(9)	T $\neg r$	[3]	X	(18)	F r	[12]	X
			(10)	F r	[9]	X		* 17 Re	X	
							(19)	T p	[12]	X
								* 16 19	X	

This tableau might be proving (once correct):

$\vdash (((\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p)) \wedge ((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r)))$

Problems

- (18) Second close reference is invalid.

Prettify formulas

Print

Export as JSON

Import from JSON

Help

Use $\&$, \wedge or $\&$ for conjunction, \vee , \vee or \vee for disjunction, \rightarrow or \rightarrow for implication, and \neg , \neg or \neg for negation. Conjunction and disjunction are strictly binary. Each node of the tableau contains a signed formula, i.e. it must be prefixed by T or F.

To enter a premise / assumption (which you want to prove), make it reference itself (i.e. "(1) F ... [1]").

α -rules

T $(A \wedge B)$

F $(A \vee B)$

F $(A \rightarrow B)$

T $\neg A$

F $\neg A$

(1) $F(((\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p)) \wedge ((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r)))$		[1]	x
(2)	$F((\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p))$	[1]	x
(3)	$T(\neg p \rightarrow \neg r)$	[2]	x
(4)	$F(r \rightarrow p)$	[2]	x
(5)	$T r$	[4]	x
(6)	$F p$	[4]	x
(7)	$F p$	[3]	x
(8)	$T p$	[7]	x
	* 8 6 x		
(9)	$T \neg r$	[3]	x
(10)	$F r$	[9]	x
	* 5 10 x		
(11)	$F((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r))$	[1]	x
(12)	$T(r \rightarrow p)$	[11]	x
(13)	$F(\neg p \rightarrow \neg r)$	[11]	x
(14)	$T \neg p$	[13]	x
(15)	$F \neg r$	[13]	x
(16)	$F p$	[14]	x
(17)	$T r$	[15]	x
(18)	$F r$	[12]	x
	* 17 18 x		
(19)	$T p$	[12]	x
	* 16 19 x		

This tableau might be proving (once correct):

$\vdash (((\neg p \rightarrow \neg r) \rightarrow (r \rightarrow p)) \wedge ((r \rightarrow p) \rightarrow (\neg p \rightarrow \neg r)))$

Problems

- (8) Is not an α -subformula of (7).
- (8) Closing formulas are not complementary.

Prettify formulas

Print

Export as JSON

Import from JSON

-prettify (u)

-validácia Alfy a Bety

-export/import v JSONe

-záver

-save as pdf

-helper section

-pridavanie nodov len na konci (u)

-mazanie celého podstromu (u)

ČO JE V PÔVODNOM?

ČO ROBÍM NAVYŠE?

(1)	T forall x (dieta(x) -> darcek(x))	[1] E
(2)	T E x dieta(x)	[2] E
(3)	F ve x darcek(x)	[3] E
(4)	T dieta(Saska)	Substituting Saska for x [2] E

This tableau does not prove:

$\forall x(\text{dieta}(x) \rightarrow \text{darcek}(x)) , \exists x \text{dieta}(x) \vdash \exists x \text{darcek}(x)$

Prettify formulas Print Export as JSON Import from JSON

Help

Use \wedge , \vee or \wedge for conjunction, \perp , \vee or \vee for disjunction, \Rightarrow or \rightarrow for implication, and \neg , \neg or \neg for negation. Conjunction and disjunction are strictly binary. Each node of the tableau contains a signed formula, i.e. it must be prefixed by T or F .

To write first order logic terms use ' \forall ', ' \exists ', ' \forall forall', ' \exists ' and ' \exists ', ' \exists ' quantifiers.

To enter a premise / assumption (which you want to prove), make it reference itself (i.e. "(1) F ... [1]").

α -rules

T (A \wedge B)	F (A \vee B)	F (A \rightarrow B)	T \neg A	F \neg A
T A	F A	T A	F A	T A
T B	F B	F B		

β -rules

F (A \wedge B)	T (A \vee B)	T (A \rightarrow B)
F A F B	T A T B	F A T B

γ -rules

T \forall x P(x)	F \exists x P(x)
T P(x)	F P(x)

δ -rules - use completely new variable x in subformula

F \forall x P(x)	T \exists x P(x)
F P(x)	T P(x)

-prerábka štruktúry

-renderovanie v divoch

-substitucia (transformacia a validacia)

-validacia Gammy a Delty (u)

-pridanie nodu hocikde (u)

-mazanie vybraného nodu ale aj podstromu (u)

-switch biet (u)

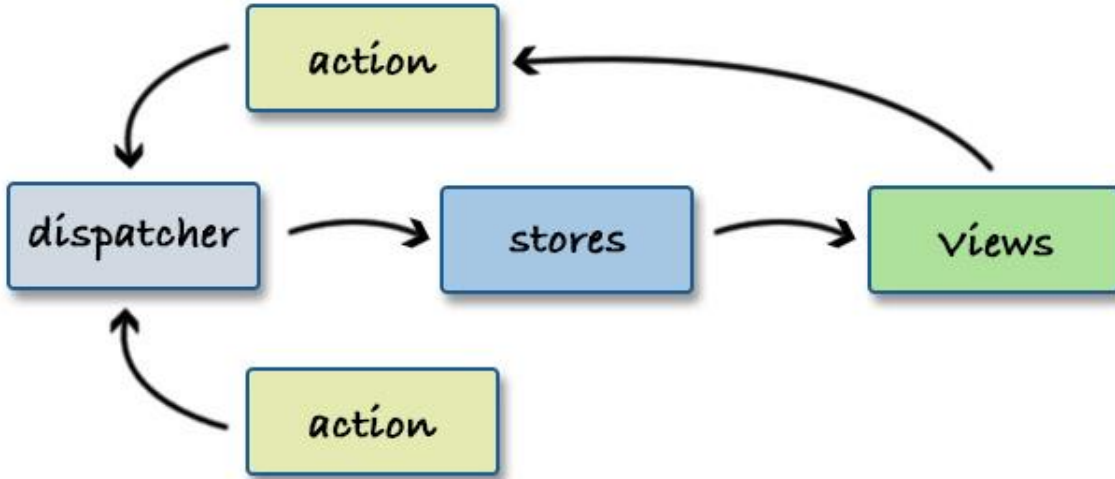
-zmena bety na alfu ak jeden z podstromov chýba (u)

-undo/redo

Explore History (27)

Import / Export

FLUX FLOW V ELME



```
update : Msg -> Model -> ( Model, Cmd Msg )
update msg model =
  case msg of
    JsonSelected ->
      ( { model | jsonImportError = "", jsonImporting = True }, fileSelected model.jsonImportId )
    - ->
      ( simpleUpdate msg { model | jsonImportError = "" }, Cmd.none )
```

```
20 main : Program Never Model Msg
21 main =
22   Html.program
23     { init = init
24       , update = update
25       , view = view
26       , subscriptions = subscriptions
27     }
28
29
30 type alias Model =
31   { tableau : Tableau
32     , jsonImporting : Bool
33     , jsonImportError : String
34     , jsonImportId : String
35   }
36
37
38 init : ( Model, Cmd msg )
39 init =
40   ( { tableau =
41     { node =
42       { id = 1
43         , value = ""
44         , reference = { str = "1", up = Just 0 }
45         , formula = Formula.parseSigned ""
46         , gui = defGUI
47       }
48     , ext = Open
49     }
50     , jsonImporting = False
51     , jsonImportError = ""
52     , jsonImportId = "importJson"
53   }
54   , Cmd.none
55 )
56
57
58 subscriptions : Model -> Sub Msg
59 subscriptions model =
60   fileContentRead JsonRead
61
```

```

173 view : Model -> Html Msg
174 view model =
175   div [ class "tableau" ]
176     [ viewNode (Zipper.zipper model.tableau)
177       , verdict model.tableau
178       , problems model.tableau
179       , p [ class "actions" ]
180         [ button [ onClick Prettify ] [ text "Prettify formulas" ]
181           , button [ attribute "onClick" "javascript:window.print()" ] [ text "Print" ]
182           , jsonExportControl model.tableau
183           , jsonImportControl model
184         ]
185       , jsonImportError model
186       , Rules.help
187     ]

```

```

104 simpleUpdate : Msg -> Model -> Model
105 simpleUpdate msg model =
106   Debug.log "model"
107     (case msg of
108     ChangeText z new ->
109       { model | tableau = z |> Zipper.setFormula new |> top }
110
111     ExpandAlpha z ->
112       { model | tableau = z |> Zipper.extendAlpha |> topRenumbered }
113
114     ExpandBeta z ->
115       { model | tableau = z |> Zipper.extendBeta |> topRenumbered }
116
117     ExpandGamma z ->
118       { model | tableau = z |> Zipper.extendGamma |> topRenumbered }
119
120     ExpandDelta z ->
121       { model | tableau = z |> Zipper.extendDelta |> topRenumbered }
122
123     ChangeRef z new ->
124       { model | tableau = z |> Zipper.setRef new |> top }
125
126     Delete z ->
127       { model | tableau = z |> Zipper.delete |> topRenumbered }
128
129     DeleteMe z ->
130       { model | tableau = z |> Zipper.deleteMe |> topRenumbered }
131
132     MakeClosed z ->
133       { model | tableau = z |> Zipper.makeClosed |> top }
134
135     SetClosed which z ref ->
136       { model | tableau = z |> Zipper.setClosed which ref |> top }
137
138     MakeOpen z ->
139       { model | tableau = z |> Zipper.makeOpen |> top }
140
141     ChangeVariable z newVariable ->

```

ĎAKUJEM ZA POZORNOST

Otázky

GREAT QUESTION



**FIND THE
ANSWER & TELL THE CLASS TOMORROW**