Lógica

aula 19: A outra face das regras de Hilbert

Liduína já tinha mesma aprendido muita coisa.

Ela tinha aprendido a usar o Modus Ponens.

E tinha aprendido a usar as regras `H1` e `H2` para construir coisas como

$$\mathtt{A} \;\to\; (\mathtt{A} \to \mathtt{A})$$

$$\big(\mathtt{A} \;\to\; (\mathtt{A} \to \mathtt{A})\big) \;\to\; \Big(\mathtt{A} \;\to\; \big(\mathtt{A} \to (\mathtt{A} \to \mathtt{A})\big)\Big)$$

$$\big(\mathtt{A} \;\to\; (\mathtt{A} \to \mathtt{A})\big) \;\to\; \big((\mathtt{A} \to \mathtt{A}) \;\to\; (\mathtt{A} \to \mathtt{A})\big)$$

E ela tinha aprendido que aplicando o Modus Ponens sobre as coisas que ela tinha construído (e construindo coisas para poder aplicar o Modus Ponens depois), ela encontrava novas maneiras de construir as coisas.

Quer dizer, foi desse jeito que ela acabou construindo coisas como

$$\mathtt{A} \;\to\; \big(\mathtt{A} \to (\mathtt{A} \to \mathtt{A})\big) \qquad\qquad (\mathtt{A} \to \mathtt{A}) \;\to\; (\mathtt{A} \to \mathtt{A})$$

$$(\mathtt{A} \to \mathtt{A}) \;\to\; \big(\mathtt{A} \to (\mathtt{A} \to \mathtt{A})\big) \qquad\qquad \mathtt{A} \;\to\; \mathtt{A}$$

Liduína já estava começando a entender que o jogo lógico de Hilbert era um *jogo de construção*.

Quer dizer, um jogo onde a gente constrói novas coisas montando, desmontando e remontando as coisas que a gente já tem.

No caso, as regras `H1` e `H2` são *regras de montagem*.

Enquanto que o Modus Ponens é uma *regra de desmontagem*.

Mas, o jogo lógico de Hilbert não é um jogo onde a gente pode construir qualquer coisa.

Era aí que estava a esperteza do Hilbert ...

O que Liduína já tinha descoberto também é que as coisas que a gente consegue construir no jogo de Hilbert são coisas (ou perguntas) que tem a resposta *"Sim, claro!"*.

Quer dizer, coisas que a gente consegue descobrir raciocinando apenas logicamente, sem saber nada sobre nada.

Mas tinha coisas que Liduína ainda não tinha entendido ...

1. *Eu não to entendendo é nada!*

   Quer dizer, eu posso até ter aprendido a fazer um monte de coisa.

   Mas entender mesmo, eu ainda não entendi foi coisa nenhuma.

   E o pior é que, se eu não consigo nem entender as regrinhas do Hilbert, como é que eu vou querer entender aquilo que está por trás das regrinhas ...

   `hmm...        :/`

   Mas, peraí ...

   Eu acho que eu to meio mal-humarada hoje ...

   Alguma coisa eu já consegui entender sim.

   Quer dizer, eu acho que eu entendi pra que serve a regrinha H1.

   Ela serve pra gente colocar qualquer coisa na frente das coisas que a gente já construiu.

   E, eu acho que eu já entendi como é que a regrinha H1 *funciona também.*

   Quer dizer, é só olhar

   $$\texttt{[H1]} \quad \mathtt{p} \ \rightarrow \ (\mathtt{q} \rightarrow \mathtt{p})$$

   *De um lado eu tenho uma coisa (*p*).*

   *E do outro lado eu tenho essa mesma coisa (*p*), com uma outra coisa qualquer (*q*) na frente.*

   O Modus Ponens me diz que eu posso usar essa regrinha se eu já tenho o p.

   E daí, então, ele me dá o $\mathtt{q} \rightarrow \mathtt{p}$.

   Hmm ...

   É como se fosse assim

   $$\mathtt{p} \quad \xRightarrow{\texttt{H1+MP}} \quad \mathtt{q} \rightarrow \mathtt{p}$$

   Quer dizer, é como se H1 *(junto como o MP) estivesse transformando o* p *no* $\mathtt{q} \rightarrow \mathtt{p}$.

   hi hi hi

   Entendi!

2. *Ou melhor, entendi o jeito que eu posso usar pra tentar entender a regrinha* H2 *também.*

   Quer dizer, é só olhar

   $$\texttt{[H2]} \quad \big(p \ \rightarrow \ (q \rightarrow r)\big) \ \rightarrow \ \big(\,(p \rightarrow q) \ \rightarrow \ (p \rightarrow r)\,\big)$$

   E daí a gente vê que (junto com o Modus Ponens) a regrinha H2 *faz a seguinte transformação:*

   $$\mathtt{p} \ \rightarrow \ (\mathtt{q} \rightarrow \mathtt{r}) \quad \xRightarrow{\texttt{H2+MP}} \quad (\mathtt{p} \rightarrow \mathtt{q}) \ \rightarrow \ (\mathtt{p} \rightarrow \mathtt{r})$$

Mas, olhando bem direitinho para essa transformação, a gente vê que ela é uma coisa bem conhecida.

Quer dizer, isso é que nem a operação de distributividade da matemática

$$3 * \big(5 + 2\big) \quad = \quad 3*5 + 3*2$$

Hmm ...

To entendendo ...

Dizem que o Hilbert gostava de matemática também ...

E, pelo jeito, ele está fazendo lógica que nem se faz matemática ...

Hmm ...

Isso parece uma boa pista ...

3. *Mas, deixa eu voltar pra lógica.*

   Agora eu já entendo como a regra H2 *funciona.*

   E eu já entendo pra que ela serve também.

   Então tá na hora de colocar esse entendimento em prática.

   Deixa eu ver ...

   Eu consigo construir isso aqui com a regra H1

$$\mathtt{A} \;\rightarrow\; (\mathtt{B} \rightarrow \mathtt{A})$$

   e daí eu posso aplicar a regra H2 *pra transformar isso em*

$$(\mathtt{A} \rightarrow \mathtt{B}) \;\rightarrow\; (\mathtt{A} \rightarrow \mathtt{A})$$

   Quer dizer, a mesma coisa também pode ser feita em dois passos.

   Primeiro eu construo uma regra usando H2 *com $A \rightarrow (B \rightarrow A))$ no lado esquerdo*

$$\big(\mathtt{A} \;\rightarrow\; (\mathtt{B} \rightarrow \mathtt{A})\big) \;\;\rightarrow\;\; \big((\mathtt{A} \rightarrow \mathtt{B}) \;\rightarrow\; (\mathtt{A} \rightarrow \mathtt{A})\big)$$

   E depois eu aplico Modus Ponens (porque eu já tenho o lado esquerdo) para obter

$$(\mathtt{A} \rightarrow \mathtt{B}) \;\rightarrow\; (\mathtt{A} \rightarrow \mathtt{A})$$

   Legal!

   A coisa funciona.

   Agora, deixa eu tentar um exemplo mais interessante.

4. Essa regra aqui faz sentido

$$(\texttt{A} \to \texttt{B}) \ \to \ \big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)$$

Quer dizer, suponha que eu sei que A *implica* B.

Então, se eu supor também que C *implica* A*, então eu posso concluir que* C *implica* B.

Faz sentido.

Porque de C *eu pulo pra* A*, e de* A *eu pulo pra* B.

Legal.

Eu consigo ver que isso está certo raciocinando logicamente.

Então, isso deve ser uma coisa que eu consigo construir usando as regrinhas do Hilbert.

Deixa eu ver ...

(1) *Esse lado direito me parece familiar*

$$(\texttt{A} \to \texttt{B}) \ \to \ \underbrace{\big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)}_{*}$$

Ele tem o formato das coisas construídas pela regra H2.

Examinando essa parte, eu consigo descobrir quem faz o papel de p, q *e* r*:*

$$(\underbrace{\texttt{C}}_{\texttt{p}} \to \underbrace{\texttt{A}}_{\texttt{q}}) \to (\underbrace{\texttt{C}}_{\texttt{p}} \to \underbrace{\texttt{B}}_{\texttt{r}})$$

E daí é fácil escrever a regra que é construída usando H2

$$\big(\texttt{C} \ \to \ (\texttt{A} \to \texttt{B})\big) \ \to \ \big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)$$

(2) *Isso aqui ainda não é bem o que eu quero.*

Quer dizer, o lado esquerdo não é esse.

Mas então, eu posso por o lado esquerdo que eu preciso no lado esquerdo disso aqui

$$(\texttt{A} \to \texttt{B}) \ \to \ \Big(\big(\texttt{C} \ \to \ (\texttt{A} \to \texttt{B})\big) \ \to \ \big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)\Big)$$

usando o truque de transformação da regrinha H1.

(3) *E daí eu distribuo*

$$\Big((\texttt{A} \to \texttt{B}) \ \to \ \big(\texttt{C} \ \to \ (\texttt{A} \to \texttt{B})\big)\Big) \ \to \ \Big((\texttt{A} \to \texttt{B}) \ \to \ \big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)\Big)$$

usando o truque de transformação da regrinha H2.

he he

Eu já to vendo a coisa que eu quero do lado direito ...

(4) *Então, agora, eu só preciso me livrar do lado esquerdo.*

Mas isso que tá do lado esquerdo é uma coisa que eu posso construir

$$\texttt{(A} \to \texttt{B)} \;\to\; \big(\texttt{C} \;\to\; \texttt{(A} \to \texttt{B)}\big)$$

usando a regra `H1`.

Então, agora, basta eu aplicar Modus Ponens para obter

$$\texttt{(A} \to \texttt{B)} \;\to\; \big(\texttt{(C} \to \texttt{A)} \;\to\; \texttt{(C} \to \texttt{B)}\big)$$

que é a coisa que eu queria.

Nesse ponto Liduína achou que era a hora de fazer uma pausa para organizar as suas ideias.

Ela tinha descoberto que era possível entender o funcionamento das regrinhas `H1` e `H2` em termos de operações de transformação.

Quer dizer, ela viu que a regra `H2` podia ser utilizada (junto com o Modus Ponens) para realizar a operação de distributividade

$$\texttt{p} \;\to\; \texttt{(q} \to \texttt{r)} \quad \xRightarrow{\texttt{Dist}} \quad \texttt{(p} \to \texttt{q)} \;\to\; \texttt{(p} \to \texttt{r)}$$

Depois, examinando o raciocínio que ela tinha acabado de fazer, ela viu que a regrinha `H1` estava sendo usada de duas maneira diferentes:

- no passo (2) ela foi usada para <u>colocar</u> uma coisa no lado esquerdo de uma outra coisa (que já tinha sido construída)
- no passo sf (4) ela foi usada para <u>descartar</u> uma coisa (que podia ser construída) do lado esquerdo de uma outra coisa

Pensando um pouquinho no assunto, ela lembrou que isso era o mesmo comportamento das regras `R2` e `R3` do Ubiratan.

Quando ela percebeu isso, ela resolveu escrever as coisas assim

$$\texttt{p} \quad \xRightarrow{\texttt{R2}} \quad \texttt{q} \;\to\; \texttt{p}$$

e

$$\texttt{q} \to \texttt{p} \quad \xRightarrow{\texttt{R3}} \quad \texttt{p} \qquad \text{, se } \texttt{q} \text{ pode ser construído}$$

E daí ela viu que o seu argumento podia ser escrito de uma forma bem arrumadinha

$$\xRightarrow{\texttt{H2}} \quad \big(\mathtt{C} \to (\mathtt{A} \to \mathtt{B})\big) \;\to\; \big((\mathtt{C} \to \mathtt{A}) \to (\mathtt{C} \to \mathtt{B})\big)$$

$$\xRightarrow{\texttt{R2}} \quad (\mathtt{A} \to \mathtt{B}) \;\to\; \Big(\big(\mathtt{C} \to (\mathtt{A} \to \mathtt{B})\big) \;\to\; \big((\mathtt{C} \to \mathtt{A}) \to (\mathtt{C} \to \mathtt{B})\big)\Big)$$

$$\xRightarrow{\texttt{Dist}} \quad \Big((\mathtt{A} \to \mathtt{B}) \;\to\; \big(\mathtt{C} \to (\mathtt{A} \to \mathtt{B})\big)\Big) \;\to\; \Big((\mathtt{A} \to \mathtt{B}) \;\to\; \big((\mathtt{C} \to \mathtt{A}) \to (\mathtt{C} \to \mathtt{B})\big)\Big)$$

$$\xRightarrow{\texttt{R3}} \quad (\mathtt{A} \to \mathtt{B}) \;\to\; \big((\mathtt{C} \to \mathtt{A}) \to (\mathtt{C} \to \mathtt{B})\big)$$

Liduína gostou dessa maneira de escrever as coisas, porque assim dava para ver a estrutura de alto nível do seu raciocínio.

E daí ela resolveu seguir adiante ...

5. *Deixa eu ver esse exemplo aqui*

$$\big(\mathtt{A} \;\to\; (\mathtt{A} \to \mathtt{B})\big) \;\to\; (\mathtt{A} \to \mathtt{B})$$

(1) *Hmm ...*

Só de bater o olho eu já vejo que isso pode ter sido o resultado de uma distribuição.

Quer dizer, isso pode ter vindo daqui:

$$\mathtt{A} \;\to\; \big((\mathtt{A} \to \mathtt{B}) \;\to\; \mathtt{B}\big)$$

(2) *Hmm ...*

E isso aqui, daonde terá vindo?

Deixa eu ver primeiro se isso faz sentido ...

hi hi hi

Já vi!

Isso é o Modus Ponens!!

Quer dizer, isso tá dizendo que se eu tenho o $\mathtt{A}$*, então eu posso usar a regra* $(\mathtt{A} \to \mathtt{B})$ *pra conseguir o* $\mathtt{B}$.

Esse negócio de lógica é engraçado mesmo ...

A regra que a gente aplica pra poder usar as outras regras, agora é mais uma regra que eu posso usar aplicando ela mesma (quer dizer, o Modus Ponens) ...

Não é a toa que o pessoal fica meio doido ...

he he

(3) *Mas, voltando ao meu raciocínio, o contrário também é verdade.*

Quer dizer, se eu tenho a regra $(\mathtt{A} \to \mathtt{B})$*, então eu posso usar o* $\mathtt{A}$ *pra conseguir* $\mathtt{B}$.

E, nesse sentido, isso aqui também é o Modus Ponens

$$(\mathtt{A} \to \mathtt{B}) \;\to\; (\mathtt{A} \to \mathtt{B})$$

Mas, isso aqui é uma coisa que eu já sei como construir, utilizando a regra R1*:*

$$\stackrel{\texttt{R1}}{\Longrightarrow} \quad (\texttt{A} \to \texttt{B}) \ \to \ (\texttt{A} \to \texttt{B})$$

Então, eu posso usar isso como ponto de partida, pra tentar conseguir chegar em

$$\texttt{A} \ \to \ \big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{B}\big)$$

O que faz todo sentido, pois essas duas regras são só dois jeitos diferentes de escrever o Modus Ponens.

Certo.

(4) *Deixa eu ver isso aqui então*

$$(\texttt{A} \to \texttt{B}) \ \to \ (\texttt{A} \to \texttt{B})$$

Hmm ...

O que será que acontece se eu aplicar a operação de distribuição?

$$\big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{A}\big) \ \to \ \big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{B}\big)$$

Olha só!

O lado direito já tá bem pertinho daquilo que eu quero ...

(5) *Hmm ...*

Mas ainda falta um A *na frente ...*

E eu ainda preciso descartar esse lado esquerdo ...

Esse lado esquerdo aí, se tivesse um A *na frente, eu conseguia descartar*

$$\texttt{A} \ \to \ \big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{A}\big)$$

Quer dizer, isso é um negócio que dá pra construir com a regra H1*.*

E daí, é só aplicar a regra R3 *pra jogar o lado esquerdo fora.*

Peraí ...

(6) *Se tivesse um* A *na frente dos dois lados, então os meus problemas estariam resolvidos*

$$\Big(\texttt{A} \ \to \ \big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{A}\big)\Big) \ \to \ \Big(\texttt{A} \ \to \ \big((\texttt{A} \to \texttt{B}) \ \to \ \texttt{B}\big)\Big)$$

Mas, se eu não to enganada, é exatamente isso que a regra anterior faz

$$(\texttt{A} \to \texttt{B}) \ \to \ \big((\texttt{C} \to \texttt{A}) \ \to \ (\texttt{C} \to \texttt{B})\big)$$

Pra eu ver isso direito, deixa eu trocar tudo por `p`, `q` *e* `r`

$$(p \to q) \;\to\; \big((r \to p) \;\to\; (r \to q)\big)$$

E agora deixa eu colocar isso no formato de uma transformação

$$p \;\to\; q \quad\Longrightarrow\quad (r \to p) \;\to\; (r \to q)$$

Isso aqui é a transformação que coloca alguma coisa na frente dos dois lados de uma regra.

É exatamente o que eu preciso!

Mas, como é que eu uso essa transformação?

Hmm ...

Deixa eu pensar ...

É só eu seguir o mesmo padrão da operação de distribuição

$$\texttt{Dist} \quad = \quad \texttt{H2} \;+\; \texttt{MP}$$

Quer dizer, eu realizo a distribuição construindo primeiro alguma coisa com a regra `R2`, *e depois aplicando Modus Ponens.*

Mas, no meu caso, o que eu tenho é a regra

$$(p \to q) \;\to\; \big((r \to p) \;\to\; (r \to q)\big)$$

E eu já tenho o lado esquerdo

$$\big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{A}\big) \;\to\; \big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{B}\big)$$

que é aquilo que eu quero botar um `A` *na frente dos dois lados.*

Então, eu consigo descobrir que é o `p` *e o* `q`

$$\big(\underbrace{(\texttt{A} \to \texttt{B}) \;\to\; \texttt{A}}_{\texttt{p}}\big) \;\to\; \big(\underbrace{(\texttt{A} \to \texttt{B}) \;\to\; \texttt{B}}_{\texttt{q}}\big)$$

e $\texttt{r} = \texttt{A}$, *claro.*

E a regra fica assim

$$\Big(\big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{A}\big) \;\to\; \big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{B}\big)\Big)$$
$$\to \quad \Big(\texttt{A} \;\to\; \big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{A}\big)\Big) \;\to\; \Big(\texttt{A} \;\to\; \big((\texttt{A} \to \texttt{B}) \;\to\; \texttt{B}\big)\Big)$$

O próximo passo é aplicar Modus Ponens (porque eu já tenho o lado esquerdo) para obter

$$\Big(\mathtt{A} \ \rightarrow \ \big((\mathtt{A} \rightarrow \mathtt{B}) \ \rightarrow \ \mathtt{A}\big)\Big) \ \rightarrow \ \Big(\mathtt{A} \ \rightarrow \ \big((\mathtt{A} \rightarrow \mathtt{B}) \ \rightarrow \ \mathtt{B}\big)\Big)$$

E agora os meus problemas estão resolvidos!

Quer dizer, eu posso descartar o lado esquerdo, porque isso é uma coisa que eu consigo construir com a regra H1

$$\stackrel{\mathtt{R3}}{\Longrightarrow} \quad \mathtt{A} \ \rightarrow \ \big((\mathtt{A} \rightarrow \mathtt{B}) \ \rightarrow \ \mathtt{B}\big)$$

E daí eu faço a distribuição para conseguir aquilo que eu queria desde o início

$$\stackrel{\mathtt{Dist}}{\Longrightarrow} \quad \big(\mathtt{A} \ \rightarrow \ (\mathtt{A} \rightarrow \mathtt{B})\big) \ \rightarrow \ (\mathtt{A} \rightarrow \mathtt{B})$$

Liduína estava morta de feliz.

Ela tinha acabado de inventar a sua própria transformação.

E estava se sentindo o próprio Hilbert de saias ...

Para comemorar o seu feito, ela resolveu batizar a sua transformação com o nome de L1

$$p \ \rightarrow \ q \quad \stackrel{\mathtt{L1}}{\Longrightarrow} \quad (r \rightarrow p) \ \rightarrow \ (r \rightarrow q)$$

E depois ela reescreveu o seu argumento da seguinte maneira

$$\stackrel{\mathtt{R1}}{\Longrightarrow} \quad (A \rightarrow B) \ \rightarrow \ (A \rightarrow B)$$

$$\stackrel{\mathtt{Dist}}{\Longrightarrow} \quad \big((A \rightarrow B) \ \rightarrow \ A\big) \ \rightarrow \ \big((A \rightarrow B) \ \rightarrow \ B\big)$$

$$\stackrel{\mathtt{L1}}{\Longrightarrow} \quad \Big(A \ \rightarrow \ \big((A \rightarrow B) \ \rightarrow \ A\big)\Big) \ \rightarrow \ \Big(A \ \rightarrow \ \big((A \rightarrow B) \ \rightarrow \ B\big)\Big)$$

$$\stackrel{\mathtt{R3}}{\Longrightarrow} \quad A \ \rightarrow \ \big((A \rightarrow B) \ \rightarrow \ B\big)$$

$$\stackrel{\mathtt{Dist}}{\Longrightarrow} \quad \big(\mathtt{A} \ \rightarrow \ (\mathtt{A} \rightarrow \mathtt{B})\big) \ \rightarrow \ (\mathtt{A} \rightarrow \mathtt{B})$$

Mas, agora não ia tardar muito para Liduína descobrir mais uma transformação ...

6. *Examinando o meu raciocínio em alto nível, eu vejo que aconteceu uma coisa curiosa. Quer dizer, nessa passagem aqui*

$$(\mathtt{A} \rightarrow \mathtt{B}) \ \rightarrow \ (\mathtt{A} \rightarrow \mathtt{B}) \quad \Longrightarrow \quad \ldots \quad \Longrightarrow \quad \mathtt{A} \rightarrow \big((\mathtt{A} \rightarrow \mathtt{B}) \rightarrow \mathtt{B}\big)$$

aconteceu uma inversão nas posições do $(\mathtt{A} \rightarrow \mathtt{B})$ *e do* A.

Talvez isso seja uma coisa que eu sempre consigo fazer.

Deixa eu ver ...

Imagina que eu tenho uma coisa assim

$$\mathtt{X} \ \rightarrow \ (\mathtt{Y} \rightarrow \mathtt{Z})$$

e eu queria inverter as posições do X *e do* Y *pra ficar assim*

$$\mathtt{Y} \ \rightarrow \ (\mathtt{X} \rightarrow \mathtt{Z})$$

Certo.

Então eu devia refazer os passos que eu fiz no raciocínio agora pouco.

Quer dizer, primeiro eu aplico Dist

$$\mathtt{X} \ \rightarrow \ (\mathtt{Y} \rightarrow \mathtt{Z}) \quad \xRightarrow{\mathtt{Dist}} \quad (\mathtt{X} \rightarrow \mathtt{Y}) \ \rightarrow \ (\mathtt{X} \rightarrow \mathtt{Z})$$

E depois ...

Hmm ...

Eu aplico L1 *pra colocar o* Y *na frente dos dois lados*

$$\xRightarrow{\mathtt{L1}} \quad \big(\mathtt{Y} \ \rightarrow \ (\mathtt{X} \rightarrow \mathtt{Y})\big) \ \rightarrow \ \big(\mathtt{Y} \ \rightarrow \ (\mathtt{X} \rightarrow \mathtt{Z})\big)$$

E daí, finalmente, eu aplico R3 *pra descartar o lado esquerdo, porque isso é uma coisa que eu consigo construir com a regra* H1

$$\xRightarrow{\mathtt{R3}} \quad \mathtt{Y} \ \rightarrow \ (\mathtt{X} \rightarrow \mathtt{Z})$$

Funcionou!!

7. *Eu acho que eu acabei de inventar uma nova transformação.*

   E faz sentido chamar essa transformação de Inv.

   Pelo que eu entendi, Inv *é como se fosse um programinha que executa outras 3 transformações em sequência*

$$\mathtt{Inv} \quad = \quad \mathtt{Dist} \ + \ \mathtt{L1} \ + \ \mathtt{R3}$$

   onde tudo é automático

   - *o* Dist *faz aquilo que o* Dist *sempre faz*
   - L1 *coloca* Y *na frente dos dois lados*
   - *e* R3 *sempre descarta uma coisa que tem a cara* $\mathtt{Y} \rightarrow (\mathtt{X} \rightarrow \mathtt{Y})$

   Legal!

Eu to começando a entender o jogo lógico do Hilbert.

Quer dizer, a gente pode entender o jogo de duas maneiras:

- *como um jogo de construção, onde a gente monta e desmonta coisas, pra descobrir novas maneiras de construir as coisas*
- *como um jogo de transformação, onde a gente muda as coisas umas nas outras, pra descobrir novas maneiras de transformar as coisas*

E esse jogo de transformação é parecido com programação, onde a gente pode juntar várias transformações mais simples, pra formar uma transformação mais complicada.

hi hi hi

Esse Hilbert era doido mesmo ...

E pensar que ele fez tudo isso antes do computador ser inventado ...

Hmm ...

Será que é por isso que as pessoas não entendem daonde vieram as regrinhas ...

Isso é uma outra pista!

8. *Mas, agora eu queria colocar o* `Inv` *em prática.*

   E eu acho que dá pra fazer isso nesse exemplo aqui

$$(\mathtt{A} \to \mathtt{B}) \;\to\; \big((\mathtt{B} \to \mathtt{C}) \;\to\; (\mathtt{A} \to \mathtt{C})\big)$$

   Em primeiro lugar, é fácil ver que isso faz sentido.

   Quer dizer, se eu assumo que `A` *implica* `B`*, e eu assumo que* `B` *implica* `C`*, então é claro que* `A` *implica* `C`*.*

   Ou, é lógico.

   Mas, se é lógico, então isso é uma coisa que eu consigo construir.

   Deixa eu ver ...

   Eu podia aplicar `Inv` *para ver o que acontece ...*

$$(\mathtt{B} \to \mathtt{C}) \;\to\; \big((\mathtt{A} \to \mathtt{B}) \;\to\; (\mathtt{A} \to \mathtt{C})\big)$$

   he he

   Acabou!

   Isso aqui é aquele exemplo que eu fiz no ponto 4 (só que com as letras trocadas).

   Então, isso é uma coisa que eu sei construir.

   Mas então, se eu construir isso e depois aplicar `Inv`*, daí eu chego naquilo que eu queria.*

   Acabou!

9. *Quer dizer, acabou mas ainda tem mais uma coisa.*

   Eu já entendi que toda regra que eu construo pode virar uma transformação.

   Então, essa regra aqui

$$\mathtt{(A \to B)} \;\to\; \big(\mathtt{(B \to C)} \;\to\; \mathtt{(A \to C)}\big)$$

   escrita com `p, q` *e* `r`

$$\mathtt{(p \to q)} \;\to\; \big(\mathtt{(q \to r)} \;\to\; \mathtt{(p \to r)}\big)$$

   pode ser vista como essa transformação aqui

$$\mathtt{p \to q} \quad\Longrightarrow\quad \mathtt{(q \to r)} \;\to\; \mathtt{(p \to r)}$$

   que ainda não tem nome.

   Mas, o que significa isso?

   Hmm ...

   Uma transformação é a mesma coisa que aplicar Modus Ponens em uma regra.

   No caso, aplicando Modus Ponens na regra aí de cima (se eu tiver $\mathtt{p \to q}$*), eu obtenho isso*

$$\mathtt{(q \to r)} \;\to\; \mathtt{(p \to r)}$$

   Mas isso é uma coisa que eu posso aplicar Modus Ponens de novo (se eu tiver $\mathtt{q \to r}$*).*

   E nesse caso eu obtenho isso

$$\mathtt{p \;\to\; r}$$

   Hmm ...

   Então, eu posso pensar que se eu tiver isso

$$\mathtt{p \;\to\; q}$$

   e isso

$$\mathtt{q \;\to\; r}$$

   então a regra aí de cima me permite obter isso

$$\mathtt{p \;\to\; r}$$

   Eu posso escrever as coisas assim

$$\mathtt{p \to q} \quad \textit{e} \quad \mathtt{q \to r} \quad\Longrightarrow\quad \mathtt{p \to r}$$

   E agora é fácil ver que isso é a operação de <u>*transitividade*</u>*.*