De: Eduardo Ochs
Para: Depto de Ciência da Computação,
Alunos de Computação,
Depto de Ciências da Natureza,
Lista Brasileira de Lógica.

1 O curso de Matemática Discreta

Eu organizei o curso de MD para ele acontecer em três "camadas" mais ou menos em paralelo. O primeira camada era toda baseada em *calcular coisas* num sistema de objetos matemáticos que inclui números, valores de verdade (**V**, **F**), conjuntos e listas; tudo nesta camada pode ser calculado num número finito de operações praticamente sem precisar de criatividade nenhuma. Dois exemplos:

$$\begin{array}{rcl}
(\forall a \in \{2,3,5\}.a^2 < 10) & = & (a^2 < 10)[a := 2] \land \\
 & & (a^2 < 10)[a := 3] \land \\
 & & (a^2 < 10)[a := 5] \\
 & = & (2^2 < 10) \land (3^2 < 10) \land (4^2 < 10) \\
 & = & (4 < 10) \land (9 < 10) \land (16 < 10) \\
 & = & \mathbf{V} \land \mathbf{V} \land \mathbf{F} \\
 & = & \mathbf{F} \\
 & & \\
\{\, 10x + y \mid (x,y) \in \{1,2,3\}^2, x < y \,\} & = & \{12, 13, 23\}
\end{array}$$

As três primeiras listas de exercícios — "Lista 1", "Lista 2" e "Notas sobre set comprehensions", disponíveis aqui,

http://angg.twu.net/2018.2-MD/20180813_lista_1.pdf
http://angg.twu.net/2018.2-MD/20180813_lista_2.pdf
http://angg.twu.net/LATEX/2018-2-MD-set-compr.pdf

eram só sobre essa primeira camada, e os alunos não tiveram grandes dificuldades com elas. Note que não pus acima o passo-a-passo para calcular $\{\, 10x + y \mid (x,y) \in \{1,2,3\}^2, x < y \,\}$ — ele está explicado na lista sobre "set comprehensions".

A segunda camada é parecida com a primeira, mas ela permite conjuntos infinitos, como $\mathbb{N}$ e $\mathbb{Z}$ — por exemplo, nela podemos usar expressões como "$\exists a \in \mathbb{Z}.5a = 1000$" e "$\exists a \in \mathbb{Z}.5a = 1002$" — cuja expansão seria infinita; para calcular o valor dessas expressões precisamos de "contas" com '...' em alguns passos, e argumentos em português — e o argumento para ver que $(\exists a \in \mathbb{Z}.5a = 1000) = \mathbf{V}$ é diferente do que mostra que $(\exists a \in \mathbb{Z}.5a = 1002) = \mathbf{F}$...

O livro principal que utilizamos no curso é o Edward Scheinerman, que começa na segunda camada, definindo pares, ímpares e divisibilidade em inteiros para tentar explicar o que são *definições*, *verdade*, *proposições*, *teoremas*, *provas*, e coisas assim. A grande maioria dos nossos calouros de Computação não consegue visualizar as coisas que ele tenta explicar, e o meu modo de tentar consertar isso era recorrendo à primeira camada. Por exemplo, nós fizemos várias definições diferentes de "a divide b" — umas com infinitos testes, outras com finitos, umas certas, outras erradas — e os alunos tentaram calcular cada uma delas em várias situações para desenvolverem intuição de como elas funcionavam, e para adquirirem prática com definições, com operações booleanas e com quantificadores. *Depois que eles tinham intuição suficiente sobre a "primeira camada" eles conseguiam entender o suficiente da "segunda camada"* — e do capítulo 1 do livro.

A terceira camada do curso é sobre *demonstrações* — numa linguagem que permite passos com "suponha" e "então" e argumentos muito mais complicados do que só séries de igualdades e desigualdades. É bem difícil ensinar demonstrações para alunos de primeiro semestre hoje em dia. A maioria deles — eu chutaria algo como 80% — tenta fazer as demonstrações num português que há alguns anos atrás nós consideraríamos incompreensível, e que é tão impreciso que fica difícil apontar exatamente onde estão os erros. Um modo de contornar este problema é apresentar algum modo de fazer demonstrações passo a passo que seja fácil de ler, que seja razoavelmente próximo de como costumamos escrever demonstrações em português, e que tenha regras precisas que nos permitam identificar passos errados. Nenhum dos livros-texto que usamos no curso, i.e., nem o do Edward Scheinerman nem o da Judith Gersting, faz isso, então eu modifiquei o sistema (informal) do Scheinerman pra que ele pudesse ser visto como uma espécie de "front-end" do sistema LK do Gentzen:

> http://angg.twu.net/LATEX/2018-2-MD-sequentes.pdf
> http://angg.twu.net/LATEX/2018-2-MD-demonstracoes.pdf

Isto é um exemplo de uma demonstração no sistema "baseado no Scheinerman":

1) Suponha $a \in \mathbb{Z}$
2) Suponha $\mathsf{impar}(a)$
3) Então $\mathsf{impar}(a)$
4) Então $\exists b \in \mathbb{Z}.2b+1=a$ (Por 3, def)
5) Suponha $b \in \mathbb{Z}, 2b+1=a$
6) Então $\begin{aligned} a^2 &= (2b+1)^2 \\ &= 4b^2+4b+1 \\ &= 2(b^2+2b)+1 \end{aligned}$
7) Então $2b^2+2b \in \mathbb{Z} \land 2(b^2+2b)+1=a^2$
8) Então $\exists c \in \mathbb{Z}.2c+1=a^2$ $(c := 2b^2+2b)$ $(\exists F)$
9) Então $\mathsf{impar}(a^2)$
10) Então $\mathsf{impar}(a^2)$ (Usa 4, fecha 5) $(\exists D)$
11) Então $\mathsf{impar}(a) \to \mathsf{impar}(a^2)$ (fecha 2)
12) Então $\forall a \in \mathbb{Z}.\mathsf{impar}(a) \to \mathsf{impar}(a^2)$ (fecha 1) $(\forall D)$

As anotações como "(∃F)" e "(∀D)" à direita indica que regras usamos: para concluir a linha 8 usamos a "regra fácil do '∃' ", e para concluir a linha 12 usamos a 'regra difícil do '∀' ".

Meu plano inicial era compararmos esse dois sistemas, o baseado no Scheinerman e o LK, com a versão do sistema Fitch usada no livro do Barwise/Etchemendy ("Language, proof, and logic"), que o Fernando Naufel usou em alguns dos semestres em que ele deu Matemática Discreta, mas não deu tempo... acabamos vendo só algumas idéias do Fitch, e usando a regra de "tautological consequence" dele (na parte sobre "regras admissíveis" do curso).

2 Quadros

A página do curso era sempre atualizada para ter fotos de todos os quadros de todas as aulas pra que 1) os alunos não *precisassem* copiar os quadros e pudessem dedicar mais tempo da aula fazendo exercícios em grupo (veja a sec.3), 2) eles pudessem checar se copiaram algo errado, 3) eles tivessem pudessem imprimir os quadros — possivelmente na versão PDFizada — pra rever a matéria antes das provas e checar detalhes, como por exemplo, a sintaxe da notação matemática. (...)

`http://angg.twu.net/2018.2-MD.html`
`http://angg.twu.net/2018.2-MD/2018.2-MD.pdf`

3 Auto-correção

Eu estruturei o curso de MD pra que ele fizesse os alunos se tornarem *autônomos* o mais rápido possível — autônomos no sentido de que eles *geralmente* conseguiriam corrigir os próprios exercícios e os dos colegas. Uma das primeiras coisas que eu distribuí impressas pros alunos foi isto aqui, que eu tinha escrito alguns semestres antes pro curso de Geometria Analítica:

> 2) Cada "seja" ou "sejam" que aparece nestas folhas é uma definição, e você pode usá-los como exemplos de definições bem-escritas (ééé!!!!) pra aprender jeitos de escrever as suas definições.
>
> 3) Em "matematiquês" a gente quase não usa termos como "ele", "ela", "isso", "aquilo" e "lá" — ao invés disso a gente dá nomes curtos pros objetos ou usa expressões matemáticas pra eles cujo resultado é o objeto que a gente quer (como nas pags (...) e (...))... mas *quando a gente está discutindo problemas no papel ou no quadro* a gente pode ser referir a determinados objetos *apontando pra eles com o dedo* e dizendo "esse aqui".
>
> 4) Se você estiver em dúvida sobre o que um problema quer dizer tente escrever as suas várias hipóteses — a prática de escrever as suas idéias é o que vai te permitir aos poucos conseguir resolver coisas de cabeça.
>
> 5) Muitas coisas aparecem nestas folhas escritas primeiro de um jeito detalhado, e depois aos poucos de jeitos cada vez mais curtos. Você vai ter que aprender a completar os detalhes.

6) Alguns exercícios destas folhas têm muitos subcasos. Nos primeiros subcasos você provavelmente vai precisar fazer as contas com todos os detalhes e verificá-las várias vezes pra não errar, depois você vai aprender a fazê-las cada vez mais rápido, depois vai poder fazê-las de cabeça, e depois você vai começar a visualizar o que as contas “querem dizer” e vai conseguir chegar ao resultado graficamente, sem contas; e se você estiver em dúvida se o seu “método gráfico” está certo você vai poder conferir se o “método gráfico” e o “método contas” dão aos mesmos resultados. Exemplo: p.(...).

7) Uma solução bem escrita pode incluir, além do resultado final, contas, definições, representações gráficas, explicações em português, testes, etc. Uma solução bem escrita é fácil de ler e fácil de verificar. Você pode testar se uma solução sua está bem escrita submetendo-a às seguinte pessoas: a) você mesmo logo depois de você escrevê-la — releia-a e veja se ela está clara; b) você mesmo, horas depois ou no dia seguinte, quando você não lembrar mais do que você pensava quando você a escreveu; c) um colega que seja seu amigo; d) um colega que seja menos seu amigo que o outro; e) o monitor ou o professor. Se as outras pessoas acharem que ler a sua solução é um sofrimento, isso é mau sinal; se as outras pessoas acharem que a sua solução está claríssima e que elas devem estudar com você, isso é bom sinal. *GA é um curso de escrita matemática:* se você estiver estudando e descobrir que uma solução sua pode ser reescrita de um jeito bem melhor, não hesite — reescrever é um ótimo exercício.

4 As provas

5 As queixas

6 O trancamento e os seus efeitos