Cálculo 2 - 2022.2

Aula 1 e 3: a operação '[:=]', ou:
aqui o curso tem um buraco

Eduardo Ochs - RCN/PURO/UFF
http://angg.twu.net/2022.2-C2.html

O macaco

Você já deve ter assistido o vídeo do Mathologer sobre o "Calculus Made Easy":

http://angg.twu.net/mathologer-calculus-easy.html

Eu vou usar esse vídeo como uma espécie de mapa pra um monte de idéias importantes de curso de Cálculo 2, e o Mathologer — obs: às vezes eu vou chamar ele de Burkard, que é o nome dele... ele respondeu um e-mail meu, então vou fingir ele é meu amigo, tá =) — mas, bom, voltando: o Mathologer diz várias vezes que a gente pode treinar um macaco pra calcular derivadas, e isso vai ser uma das coisas mais importantes do meu curso de Cálculo 2. Deixa eu explicar.

Num curso tradicional de Cálculo 1 a gente faz centenas de horas de contas na mão. Aí a gente adquire muita prática nisso e a gente passa a poder fazer o papel do macaco muito bem. Depois que a gente tem essa

prática toda a gente *começa* a poder fazer também um outro papel, que é o papel na pessoa que programa o macaco e diz quais regras ele tem que seguir...

Deixa eu dar um exemplo. No trecho do vídeo que começa no 17:00 a gente vê como o macaco calcula a derivada $\left(\frac{5+\sin x}{x^3 \ln x}\right)'$ "fazendo a álgebra no piloto automático". Se a gente seguir o vídeo com bastante atenção a gente vê que o macaco não está usando só as 5 regras pra derivada que aparecem na coluna esquerda no vídeo no 16:15, que o Burkard escreve como $(f+g)' = f'+g'$, $(f-g)' = f'-g'$, $(fg)' = f'g+fg'$, $(\frac{f}{g})' = \frac{f'g-fg'}{g^2}$ e $(f(g))' = f'(g)g'$... o macaco também usa as regras que o Burkard põe na tabela na parte de cima da tela no 13:00, e no 17:35 ele calcula em separado o resultado de $(5+\sin x)'$ — que dá $\cos x$ — e no 17:50 o macaco substitui o $(5+\sin x)'$ na expressão original por $\cos x$.

Exercício 1.

Acesse o PDF do capítulo 2 do Leithold.

Entre as páginas 68 e 70 ele tem várias contas de limites feitas passo a passo, com os sinais de '=' alinhados e com justificativas à direita. Esse formato está explicado aqui:

http://angg.twu.net/LATEX/2021-2-C2-intro.pdf#page=7

Faça as contas que o Mathologer faz entre o 17:00 e o 18:15 nesse formato, com os '='s alinhados e as justificativas à direita, e com as restrições que eu vou pôr no quadro (vou PDFizar elas depois)...

Aula 3: Derivadas formais (e algumas coisas relacionadas)

Dê uma olhada na seção 3.3 do Leithold (no capítulo 3). Os teoremas 3.3.1 até 3.3.7 dele correspondem exatamente às operações de derivação que o Mathologer usa no vídeo (obs: a regra da cadeia é o teorema 3.6.1), mas cada um desses teoremas tem uma *fórmula* e as *hipóteses* necessárias pra fórmula valer. O macaco que calcula derivadas no vídeo do Mathologer no trecho entre 17:00 e 18:15 usa só as fórmulas sem checar que as hipóteses valem. O que o macaco faz é chamado de *derivação formal*, e está explicado aqui:

https://en.wikipedia.org/wiki/Formal_derivative

Em Cálculo 2 nós vamos ver várias operações que são tão difíceis que é o melhor jeito de aprendê-las, e de debugar erros nelas, é dividindo elas em várias partes. Nós acabamos de ver que "calcular derivadas (complicadas)" pode ser dividido em "aplicar fórmulas" e "checar hipóteses",

e você deve lembrar que você passou a maior parte do seu curso de Cálculo 1 só "aplicando fórmulas" sem "checar hipóteses", ou checando as hipóteses rapidinho no olho, mas sem escrever nada tipo "aqui as hipóteses fulana e beltrana valem, então blá".

Nos últimos semestres *mooooooontes* de alunos chegaram em Cálculo 2 sem saber "aplicar fórmulas" direito. Eu também vou dividir a operação de "aplicar uma fórmula" em várias partes, até chegar a uma parte — a operação '[:=]' — que é fácil de implementar num computador e que corresponde exatamente ao que o macaco que do Mathologer faz, mas que o Mathologer não explica explicitamente.

Se você tiver qualquer dificuldade com essa operação '[:=]' o melhor modo de estudá-la é estudando a matéria de Cálculo 1 e Cálculo 2 pelo Leithold e vendo como essa operação aparece implicitamente em todo lugar, mas sempre acompanhada do "checar hipóteses".

Eu costumo dividir essa operação de "aplicar fórmula" em várias operações separadas e dar um nome "padrão" pra cada uma dessas operações. Os programas de computação simbólica também fazem essa divisão em várias suboperações, porque cada suboperação corresponde a uma função diferente, e essas funções podem ser chamadas em separado. Eu vou usar a terminologia do Maxima, que é o programa de computação simbólica que eu tenho usado... o Maxima considera "substituição" e "simplificação" como operações separadas. O "Maxima Workbook" explica substituição na seção 11.4 e simplificação no capítulo 12; ele também explica "formas canônicas" nas seções 9.2 e 9.3, e eu vou considerar que "pôr uma expressão na sua forma canônica" é uma forma de simplificação. Link:

http://roland-salz.de/Maxima_Workbook.pdf

Deixa eu dar um exemplo mais básico disso. Nós conhecemos a

fórmula $2a = a + a$. Se nós *só* substituirmos todos os 'a' nesta fórmula por 10 o resultado é $2 \cdot 10 = 10 + 10$; trocar $2 \cdot 10$ por 20, ou $10 + 10$ por 20, são consideradas "simplificações".

A substituição sempre substitui *variáveis* por *expressões*.

No vídeo do Mathologer ele às vezes abrevia $f(x)$ como f, e desabrevia isso depois, e ele às vezes escreve 'df' pra "diferencial" (veja a seção 4.9 do Leithold). É muito mais difícil formalizar contas de Cálculo — ou seja, justificar formalmente cada passo delas — quando a gente permite esses truques (que são parte da "notação de Leibniz"). Na maior parte do meu curso esses truques vão ser proibidos; ele às vezes vão ser permitidos temporariamente, mas a gente sempre vai ver como tratar as contas em que eles são permitidos como "versões abreviadas" de contas em que eles são proibidos. Depois eu vou disponibilizar um monte de links sobre porque é que os matemáticos pararam de usar a notação de

Leibniz.

Exercício 2: árvores

Vai ser muito mais fácil a gente entender como o macaco derivador funciona se a gente souber tratar expressões matemáticas como árvores.

Veja os dois screenshots abaixo — eles mostram como Maxima e o Sympy representam certas expressões como árvores. *Note que as representações são diferentes!*

http://angg.twu.net/IMAGES/luatree.png
http://angg.twu.net/IMAGES/luatree-sympy.png

Pegue algumas expressões que você obteve no exercício 1 e represente elas como árvores no formato do Maxima.

Exercício 2: dicas

Lembre que nas árvores não aparecem parênteses.
Lembre que $a - b - c = (a - b) - c$.
Lembre que $a - (b - c) \neq (a - b) - c$.
Lembre que $a - (b - c)$ e $(a - b) - c$ dão árvores diferentes.
Lembre que $a^{b^c} = a^{(b^c)}$.

MUITO IMPORTANTE: lembre que na maior parte do curso a expressão 'dx' não vai poder aparecer sozinha... eu vou até me referir a ela a toda hora como "uma espécie de fecha parêntese" pra fazer as pessoas lembrarem disso.

Funções menos elementares

No vídeo o Mathologer primeiro define "funções elementares" de um jeito, e depois, a partir do 27:26, ele diz que poderíamos ter incluído nas nossas operação que criam novas funções elementares a operação que obtém a inversa de uma função... ele não dá detalhes, mas repara que se a gente puder obter a inversa de toda função a gente vai ter que poder obter a inversa de funções constantes, como por exemplo $f(x) = 1$... então provavelmente o que ele quer dizer é que a gente quer considerar como "elementares" coisas como $\sqrt{x}$, $\sqrt[3]{x}$, $\operatorname{arcsen} x$, $\arccos x$, etc...

Obs: falta LaTeXar as coisas que a gente viu no final da aula de 2022aug31! As fotos dos quadros desse dia estão aqui:
http://angg.twu.net/2022.2-C2/C2-quadros.pdf#page=3

EDOs por chutar-e-testar

Links pros exercícios de hoje:

http://angg.twu.net/LATEX/2021-2-C2-intro.pdf#page=12
http://angg.twu.net/2022.1-C2/C2-quadros.pdf#page=4
http://angg.twu.net/LATEX/2022-1-C2-VSB.pdf#page=9
http://hostel.ufabc.edu.br/~daniel.miranda/calculo/calculo.pdf#page=185

...e os exercícios da seção 5.1 do Leithold.