Maxima

O Maxima é um *CAS* (Computer Algebra System) semelhante a sistemas como o *Mathematica*, *Maple* e outros que manipulam expressões simbólicas, com a vantagem de ser um programa de código aberto. Ele tem procedimentos para operar com números e também com expressões matemáticas.

É uma ferramenta essencial para obter resolver problemas matemáticos. Entretanto, para utilizar as ferramentas deve entender o quê está fazendo, porquê está fazendo e para quê! Abaixo o linque do site para baixar o Maxima:

- Maxima download

O Máxima já vem com uma interface gráfica para usuário chamada wxMaxima. Para usuários de sistemas Linux (Ubuntu e outros) o wxMaxima está disponível diretamente nos repositórios da distribuição.

Atenção: Se você tiver problemas para executar o Maxima em Windows, talvez seja necessário configurar o seu firewall.

Maxima em 10 passos

Esse tutorial[1)] contém os 10 primeiros passos para iniciantes no Maxima poderem acompanhar alguns roteiros deste sítio. Para quem está acostumado com as linhas de comando de programação em R, fica muito mais fácil usar o Maxima [2)]

Caso queira rodar esse tutorial a partir do arquivo do código do WxMaxima, carregue o arquivo do tutorial no seu computador.

Maxima como calculadora

- 1. Vamos começar com alguns exemplos de cálculos simples! Abaixo está

uma célula de entrada com uma simples adição. Coloque o cursor sobre ele e pressionar SHIFT-ENTER

para avaliá-lo.

```
1 + 1;
```

Se você não obter quaisquer erros, o Maxima está configurado corretamente. Se você fez um erro, você deve verificar a configuração ou wxMaxima visite o website wxMaxima (http://wxmaxima.sourceforge.net/) para instruções sobre como configurar e wxMaxima Maxima corretamente!

- 2. Assumindo que você tenha resolvido seus problemas, vamos fazer mais algumas

cálculos (mais uma vez - colocar o cursor na célula de entrada abaixo e pressionar SHIFT-ENTER para avaliar o código)!

```
 5!;
% * 10;
%o1 * 100;
1 / 3;
1.0 / 3.0;
```

Na célula de entrada acima, nós mandamos cinco linhas de comandos, para o Maxima. Cada linha deve terminar com um "; " ou um "\$". No caso da linha termina com um "; ", Maxima irá mostrar o resultado da linha, enquanto os resultados das linhas terminando com o "\$" serão suprimidos. O "\$" vem a calhar quando se faz mais cálculos. Note também que o resultado de "03/01" e "1.0/3.0" diferem. Isso porque Maxima, ao contrário de programas de matriz numérica (Matlab, etc) tenta manter cálculos precisos - expressões como 1 / 3 ou sqrt (2) não são avaliadas a menos que explicitamente solicitado. Em "1.0/3.0" foi utilizado a aproximação de "Números de Ponto flutuante" (Floating_point) para o Maxima avaliar a expressão.

- 3. Podemos, no entanto, solicitar ao Maxima uma aproximação de ponto flutuante de uma expressão. Rode o script abaixo e observe os resultados.

```
sqrt(2 * %pi);
float(%);
```

Na linha "float (%);" foi utilizado o símbolo "% ". Este símbolo contém sempre o resultado da última linha avaliados. Símbolos numerados com "o" (output) como "% o1", "% o2" armazenam os resultados que aparecem quando as células de entrada (input: %i1,%i2,...) são avaliadas.

- 4. Nós também podemos armazenar, não só números, mas as expressões inteiras, em variáveis. Use "variable_name: valor;" forma para armazenar o valor em "variable_name". Execute a célula abaixo e observe.

```
radius: 10 $
height: 100 $
area: %pi * radius^2;
volume: area * height;
```

Vamos avaliar o último resultado númericamente:

```
float(%);
```

Operando funções

- 5. Até agora usamos o Maxima apenas como uma calculadora normal. Vamos agora fazer algo que não é possível em uma calculadora simples, definir e operar funções:

```
f(x) := x^2 + a$
f(5);
f(5), a = -5;

h(x):=  1 / (x^2 + a);
h(5);
h(5), a=-20;
float(%);

f1(x):= exp(x);
g1(x):= -x^2;
f1(g1(x));
```

- 6. Agora que você já aprendeu o básico, é hora para alguns exemplos gerais de matemática! Lembre-se: se você quiser saber mais sobre uma função específica, clique sobre ele e pressione F1. Resolvendo equações usando o “solve”:

```
solve(a*x^2 + b*x + c = 0, x);
```

- 7. O Maxima, além de muitas outras coisas, faz gráficos. Rode o código abaixo!

```
wxplot2d([sin(x), cos(x)], [x,0, 2*%pi]);
wxplot3d( exp(-x^2 - y^2), [x,-2,2],[y,-2,2]);
```

Enfim, derivadas

- 8. Vamos tentar uma diferenciação usando a função “diff” .

```
f(x) := x^2 $
diff(f(x), x);
g(y) := sin(y)$
g(f(x));
diff( g(f(x)) , x);
```

Sim, o Maxima sabe a “Regra de Cadeia”!

- 9. Agora que já explorou o Maxima, vamos obter algumas derivadas. A sintaxe é bastante simples: primeiro forneça a função e em seguida defina qual a variável a ser diferenciada.

```
diff(x^2, x);
diff(log(x),x);
diff(sin(x)+ cos(x),x);
diff(sin(x)/x^2, x);
```

- 10. Muitas vezes o resultado da expressão matemática da derivada parece diferente do que

calculou.

Pode ser que não seja um erro, apenas uma forma diferente de expressar a mesma função. Para simplificar expressões matemáticas no Maxima podemos usar a função ratsimp.

```
ratsimp(%);
```

Maxima: more steps

$\lim\limits_{maxima\to \infty}= \infty$

Maxima: more steps

Gostou!

- Para uma consulta rápida e novas dicas veja o site A 10 minute tutorial for solving math problems with Maxima
- Caso queira se aprofundar um pouco mais sobre o Maxima, o guia introdutório da Bruna Santos da Universidade do Porto é um ótimo começo: Maxima Bruna Santos 2009

maxima, derivada

1)

tutorial baseado na documentação oficial do programa, modificado por — *Alexandre Adalardo de Oliveira 2012/05/02 17:47*

2)

há uma opção de pacote no R (Ryacas) que faz a interface com um outro sistema de manipulação simbólica de expressões matemáticas: o yacas

From:
http://ecovirtual.ib.usp.br/ -

Permanent link:
http://ecovirtual.ib.usp.br/doku.php?id=ecovirt:roteiro:soft:tutmaxima

Last update: **2016/05/10 07:19**