

wxMaxima Kullanıcı Kılavuzu

wxMaxima, Maxima bilgisayar cebir sistemi (CAS) için bir grafik kullanıcı arayüzüdür (GUI). wxMaxima Maxima'nın tüm fonksiyonlarını kullanmaya olanak sağlar. Buna ek olarak, en yaygın kullanılan özelliklere erişmek için kısayollar ve kullanışlı özellikler içerir. Bu kılavuz, wxMaxima'yı Maxima için en popüler GUI'lerden biri haline getiren bazı özellikleri açıklamaktadır.



- [Giriş](#): wxMaxima'nın temelleri
- Uzantılar: wxMaxima'nın maxima'ya eklenen komutları
- Sorun Giderme: wxMaxima beklendiği gibi çalışmıyorsa ne yapılacaktır
- SSS: Sıkça Sorulan Sorular
- CommandLine: wxMaxima'nın desteklediği komut satırı değişkenleri

1 wxMaxima'ya Giriş

1.1. Maxima ve wxMaxima

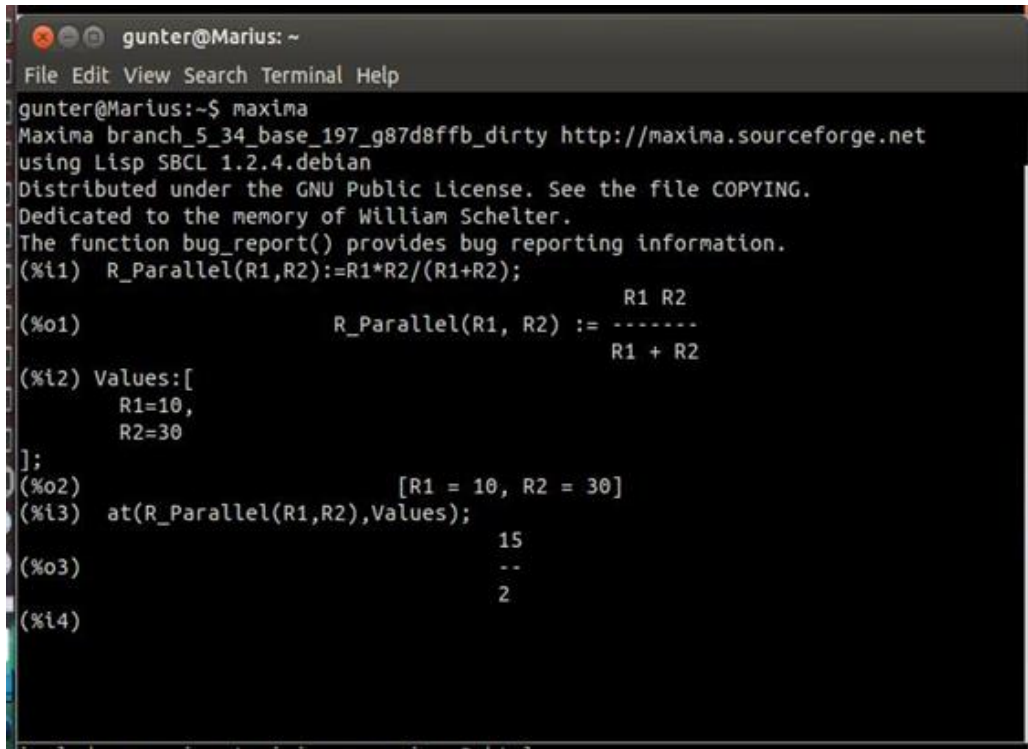
Açık kaynak denildiğinde, büyük sistemlerin küçük geliştirici grupları tarafından daha kolay olan daha küçük projeler halinde ele alınarak geliştirildiği gözlenir. Örneğin bir CD yazıcı programında, aslında CD'ye kayıt yapana bir komut satırı aracı ve kullanıcının tüm komut satırı anahtarlarını öğrenmek zorunda kalmadan ve komut satırını kullanmadan kullanmasını sağlayan bir grafik kullanıcı arayüzü bulunur. Bu yaklaşımın bir avantajı, komut satırı programının ,tekerleği yeniden keşfetmek zorunda kalmadan, pek çok program (GUI) tarafından paylaşılabilmesidir. Aynı CD yazıcısı komut satırı programı, bir dosya yöneticisi uygulaması için bir "CD'ye gönder" aracı olarak kullanılabilir, bir müzik çaların "CD'ye yaz" işlevi ve DVD yedekleme aracı bir

CD'nin yazıcısı olarak kullanılabilir . Bir başka avantaj da, büyük bir görevin daha küçük parçalara bölünmesinin, geliştiricilerin aynı program için birkaç kullanıcı arabirimi sağlamasına olanak sağlamasıdır.

Maxima gibi bilgisayar cebri sistemleri (BCS) bu çerçeveye uymaktadır. Bir BCS, hassas bir hesap makinesi uygulamasını sağlayabilir veya daha büyük bir sistemin arka planında (örneğin, SAGE) bir “matematik makinesi” olarak çalışıp formülleri otomatik olarak dönüştürebilir. Alternatif olarak, çeşitli kullanıcı arayüzleriyle kendi başına ayakta duran bir sistem olarak kullanılabilir. Maxima'ya bir komut satırı üzerinden erişilebilir. Bununla birlikte, çoğu zaman, wxMaxima gibi bir arayüz, özellikle bilgisayar cebri sistemini kullanmaya yeni başlayanlar için, daha verimli ve kolay bir yol olduğunu kanıtlıyor. .

1.1.1. Maxima

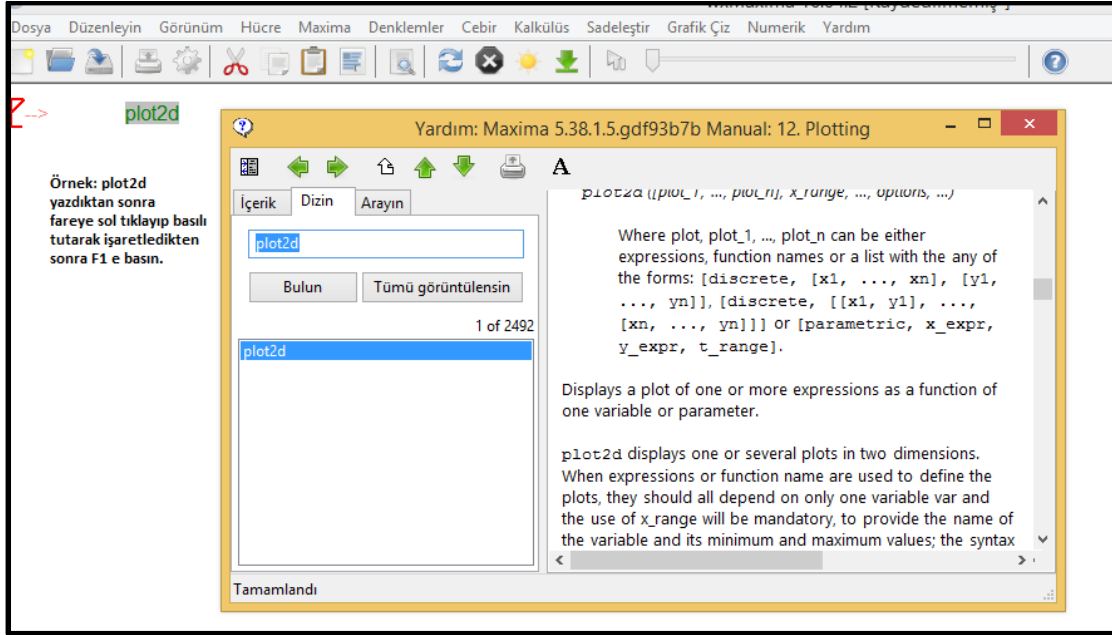
Maxima birçok özellikleri olan “tam” bir bilgisayar cebir sistemi (CAS). Diğer bir deyişle, matematiksel problemlerin sonuçlarının numerik-sayısal değerlerini üretmek yerine, bir hesap makinesinden öte, matematik problemlerini sembolik yolla çözerek, formülleri yeniden düzenleme ve yeni bir formül bulmak/geliştirmek becerisine sahip “sembolik karakterli” bir BCS dir. Ayrıca, analitik olarak çözülemeyen denklemlerin veya denklem sistemlerinin çözümü için bir dizi sayısal analiz yöntemi sunar.



```
gunter@Marius: ~  
File Edit View Search Terminal Help  
gunter@Marius:~$ maxima  
Maxima branch_5_34_base_197_g87d8ffb_dirty http://maxima.sourceforge.net  
using Lisp SBCL 1.2.4.debian  
Distributed under the GNU Public License. See the file COPYING.  
Dedicated to the memory of William Schelter.  
The function bug_report() provides bug reporting information.  
(%i1) R_Parallel(R1,R2):=R1*R2/(R1+R2);  
  
(%o1) 
$$R\_Parallel(R1, R2) := \frac{R1 R2}{R1 + R2}$$
  
  
(%i2) Values:  
      R1=10,  
      R2=30  
];  
(%o2) [R1 = 10, R2 = 30]  
(%i3) at(R_Parallel(R1,R2),Values);  
      15  
(%o3) --  
      2  
(%i4)
```

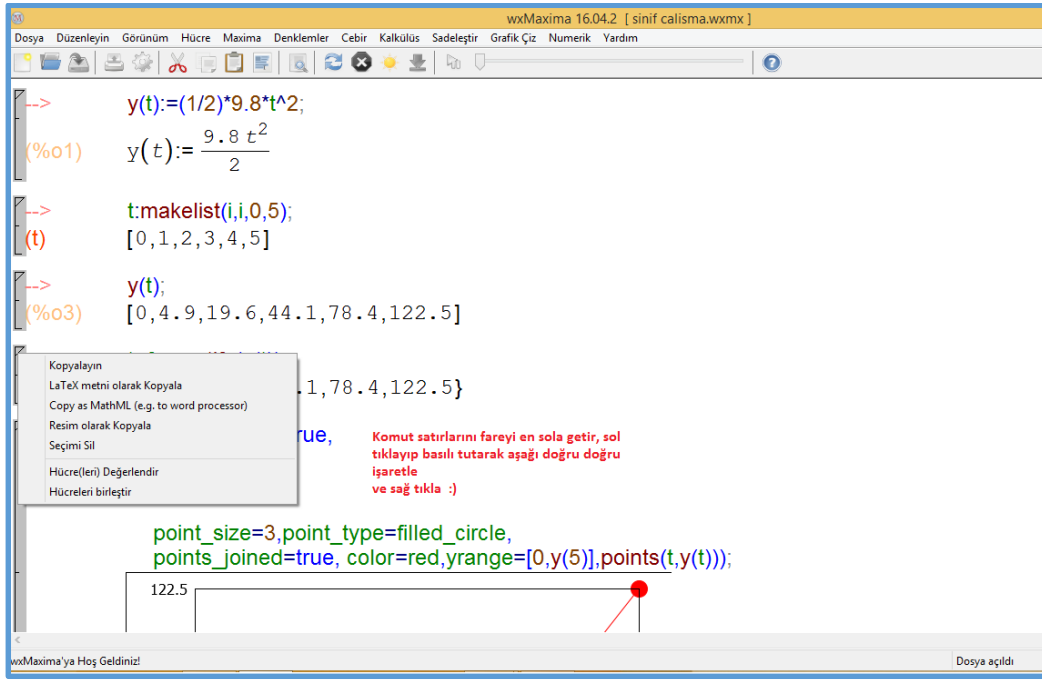
[“Maxima” BCS hakkında kapsamlı dokümanlar bulmak mümkündür.](#)

Bu dokümanların bir kısmı wxMaxima'nın yardım menüsünde de mevcuttur: <Yardım> tuşuna (çoğu sistemde <F1> tuşu olacaktır) basarak wxMaxima'nın içeriğe duyarlı yardım özelliği, otomatik olarak imleçteki komut için maxima'nın kullanım kılavuzu sayfasına atlar



1.1.2. wxMaxima

Maxima'nın tüm işlevselliğini ve esnekliğini sağlayan grafik kullanıcı arabirimidir. wxMaxima, kullanıcılara grafik ekranı ve Maxima ile çalışmayı kolaylaştıran birçok özellik sunar. Örneğin, wxMaxima, herhangi bir hücrenin içeriğini (veya gerekirse, bir formülün herhangi bir parçasını da) metin olarak, basit bir sağ tıklamayla LaTeX veya MathML belirtimi olarak dışa aktarmanızı sağlar. Gerçekten de, bir çalışma kitabı HTML dosyası olarak veya LaTeX dosyası olarak dışa aktarılabilir.



Hesaplamalar Maxima komut satırı aracı tarafından arka planda gerçekleştirilir.

1.2. wxMaxima Çalışma Sayfası

wxMaxima çoğunlukla kendini açıklar, ancak bazı ayrıntılar dikkat gerektiriyor. [Bu site](#), Maxima'nın çeşitli yönlerini ele alan bir dizi çalışma kitabını içermektedir. Bunlardan bazılarıyla (özellikle "10 dakikada Maxima " notları) çalışmak Maxima ile etkileşim kurmak için wxMaxima'nın kullanımı becerisini artıracaktır. Bu kullanım kılavuzu, açıkça görülmesi mümkün olmayan ve çevrimiçi materyalde yer almayabilecek wxMaxima'nın yönlerini tanımlama üzerine yoğunlaşmaktadır.

1.2.1. wxMaxima Çalışma Sayfası Yaklaşımı

Standart olmayan çok az şeyden biri, wxMaxima'nın Maxima'nın verilerini, yalnızca kullanıcıların bunu istediğinde değerlendirilen hürelere (Maxima'ya gönderildiği anlamına gelir) düzenlemesidir. Bir hücre değerlendirildiğinde, o hücredeki ve yalnızca o hücredeki tüm komutlar bir toplu iş olarak değerlendirilir. (Bir komşu hücre kümesini seçebilir ve bunları birlikte değerlendirebilir, Maxima'ya bir çalışma kitabındaki tüm hücreleri tek bir seferde değerlendirmesini de söyleyebiliriz.) Bu yaklaşım ilk görüşte yabancı gibi görünebilir. Ancak, büyük belgelerle çalışmayı büyük ölçüde kolaylaştırır.

Ayrıca, bu yaklaşım hata ayıklama için çok kullanışlıdır. Hata bulunan yada değişiklik yapmak istediğiniz satıra gidip değişiklikler yapabilirsiniz. wxMaxima tarafından yazılan hücreler, Maxima'ya gönderilecek olan Maxima giriş hücrelerine –Maxima input cells- dönüştürür.

wxMaxima'da her bir komut satır “;” ile biter/bitmeli. Tek bir girdi hücreğine “;” ile biten birden fazla girdi yapabilirsiniz. Bu hücreyi “shift+enter” ile değerlendirdiğinizde kaç “girdi” yaptıysanız, wxMaxima o kadar “çıkıtı” satır verir.

```
(%i1) f_101(x):=sin(x)^2;  
      g_101(x):=cos(x);  
  
(%o1) f_101(x):=sin(x)^2  
(%o2) g_101(x):=cos(x)
```

(%i1) Birinci girdi hücresi 'i' = input...
(%i1) birinci girdi satırı ";" ile ayrılmış iki fonksiyon içeriyor.
"shift+enter" ile değerlendirildiğinde
(%o1) ve (%o2) 'o' = output... çıktı
satırı alabiliyoruz.

Bir girdi hücresinin içeriğini değerlendirirken, giriş hücresi maxima girdiye bir etiket atar (varsayılan olarak kırmızı ile gösterilir ve "% i " (" i " input) tarafından tanımlanabilir olarak gösterilir), wxMaxima oturumunda daha sonra tekrar o girdi satırına giderek kullanabilirsiniz. Maxima'nın ürettiği çıktı farklı bir stilde (kullanıcı tarafından seçilebilir) gösterilecek ve % o.. ("o" output) ile başlayacak bir Maxima etiketi gelecektir. Çıktı ögesine, daha sonra bu kodu referans olarak erişilebilir. Varsayılan olarak, wxMaxima, kullanıcı tarafından bir dinamik olarak oluşturulan etiketi, daha açıklayıcı bir adla değiştirmeye çalışacaktır. Bununla birlikte, otomatik olarak üretilen Maxima'nın "% o" tarzı etiketi erişilebilir olacaktır.

Örneğin: Diyelim ki 20 satır lık bir girdi ve çıktı içeren bir wxMaxima sayfasında 20. satırdaysanız 13.satırdaki girdi yada çıktıyı kullanmanız gerekiyor. 20.satırdaysanız

- (%i13) yazarak “13.satırdaki girdinizi”
- (%o13) yazarak “13. satırdaki çıktıyı” kullanabilirsiniz.

Giriş hücrelerinin yanında, wxMaxima, doküman/belge hazırlarken, metin hücreleri, resim hücreleri, başlık hücreleri, bölüm hücreleri ve kısım hücreleri eklemek için olanak verir. Her hücrenin kendi geri alma arabelleği vardır, böylece çeşitli hücrelerin değerlerini değiştirerek hata ayıklama ve ardından gereksiz değişiklikleri yavaş yavaş geri alma işlemi daha kolaydır. Ayrıca çalışma sayfasının kendisinde, hücre düzenlemelerini geri alabilen, ekleyen ve silen bir genel geri alma arabelleği

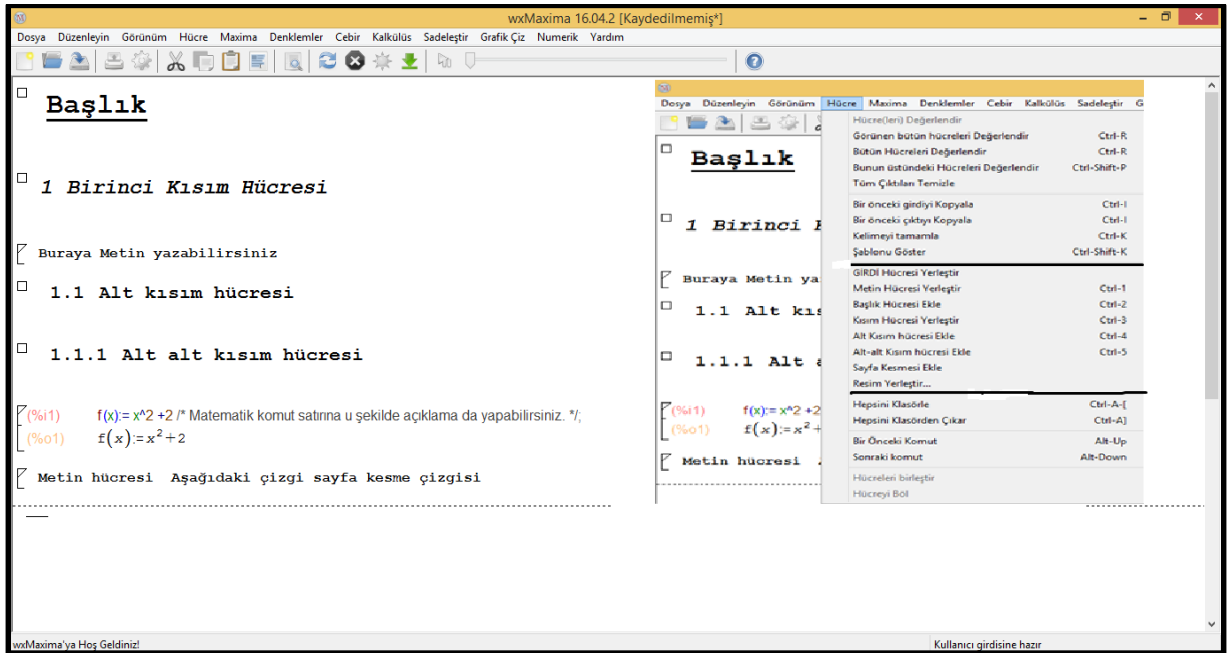
vardır. Aşağıdaki şekil bir başlık hücreini, bir bölüm hücreini, bir metin hücreini ve bir giriş / çıkış hücreinin bir bölümünü göstermektedir.

1.2.2. Hücreler

Çalışma sayfası, kelime işlemcilerin sunduğu stillere çok benzer şekilde çalışan hücrelerde düzenlenir. Her hücre, aşağıdaki içerik türlerinden birini içerebilir:

- bir veya daha fazla maxima girdi satırı
- bir şekil
- bir başlık, kısım veya alt kısım
- Maksima'dan çıktı veya bir soru
- Matematik hücresinin içeriklerinin anlamını açıklamak için kullanılabilen normal bir metin bloğu.

Metin girildiğinde wxMaxima'nın varsayılan davranışı otomatik olarak bir matematik hücresi yaratmaktır. Diğer türdeki hücreler, Hücre menüsünü kullanarak veya menüde gösterilen kısayol tuşlarını kullanarak oluşturulabilir.



1.2.3. Yatay ve Dikey İmleçler

Bazen, dışa aktarma veya Sürükle-Bırak için birden fazla hücrenin veya bir hücrenin yalnızca bir bölümünün seçilmesine izin vermek istenebilir. Bununla birlikte, bir

hücrenin ortasında böyle bir hareketi başlatmak ve bunu bir başka ortada sona erdirmek neredeyse kesinlikle beklenmedik sonuçlara neden olacaktır. wxMaxima bu durumda seçimi tam hücrelere genişletecektir.

wxMaxima, wxMaxima'nın gerektiğinde otomatik olarak geçeceği iki imleç türü tanımlayarak bu konuda çalışır:

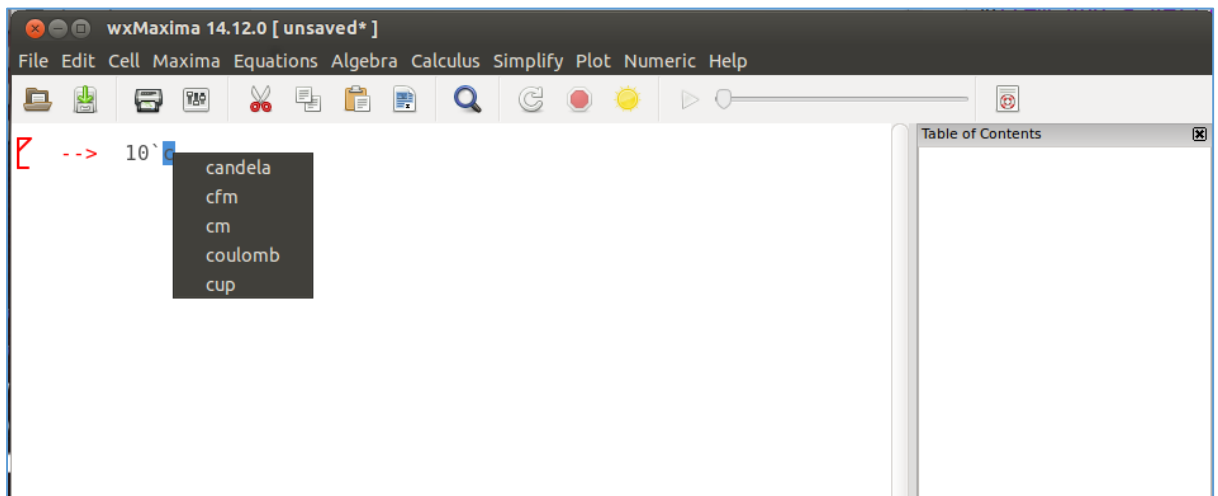
Herhangi bir sayıda tüm hücreleri seçebilen **satırlar arası çalışan** yatay bir imleç. Bu imleç, imleci iki hücre arasında hareket ettirerek veya iki hücre arasındaki boşluğa tıklayarak etkinleştirilir. Bir bitişik hücre kümesi seçmek için, üst hücrenin solundaki köşeli parantez içinde (ancak üçgene değil) tıklayın ve ardından bu hücreyi ve sonraki hücrenin (veya hücre aralığının) vurgulandığı şekilde imleci sürükleyin. Ardından, vurgulanan hücreleri yürütmek için <ctrl + enter> veya <shift + enter> tuşlarını kullanın.

Dikey imleç bir hücre içinde çalışır. Bu imleç, fare işaretçisini veya sağ-sol tuşlarını kullanarak imleci bir hücre içinde hareket ettirerek etkinleştirilir.

İmleç bir hücrenin içinde olduğu sürece, arama işlemleri kapsamalarını geçerli hücre ile sınırlar.

1.2.4. Komutların Otomatik Olarak Tamamlanması:

wxMaxima, menü aracılığıyla (Hücre / Tamamla Kelime) tıklandığında veya alternatif olarak <Ctrl> + <k> tuş kombinasyonuna basıldığında bir otomatik tamamlama özelliği içeriyor. Otomatik tamamlama içeriğe duyarlıdır ve ezUnit'ler için bir birim belirtiminde etkinleştirilirse, uygulanabilir birimlerin bir listesini sunacaktır.



Mevcut komutu veya değişken adını tamamlamanın yanı sıra, otomatik tamamlama, bu programın beklediği parametrelerin türünü (ve anlamını) gösteren komutların

çoğunda bir şablon gösterebilir. Bu özelliği etkinleştirmek için <Shift> + <Ctrl> + <k> tuşlarına basın veya ilgili menü öğesini seçin (Hücre / Şablon Göster).

1.2.5. Yunanca karakterler

Bilgisayarlar geleneksel olarak karakterleri 8 bitlik değerlerle saklar. Bu maksimum 256 farklı karaktere izin verir . Neredeyse herhangi bir dildeki tüm harfler, sayılar ve kontrol sembolleri (iletimin sonu, dizinin sonu, menüler için çizim dikdörtgenleri için çizgiler ve kenarlar vb.) Bu sınırın içine girebilir.

Çoğu ülke için, seçilen 256 karakterden oluşan kod sayfası Yunanca harf içermez. Bu sınırlamanın üstesinden gelmek için unicode icat edilmiştir: Bu, normal olarak İngilizce dilde kullanılmayan karakterleri (yalnızca Latin karakterlerinin temel biçimi kullanıldığı sürece) düz 8-bitlik ASCII'ye benzeyen bir metne ekleme yöntemidir .

Maxima, onları destekleyen bir lisp üzerinde çalışıyorsa unicode karakterlere izin verir. WxMaxima'nın üzerine kurulmuş olan wxWidgets kitaplığı da unicode karakterlerini destekliyorsa, wxMaxima unicode desteği ile oluşturulabilir. Bu durumda (bir Yunan Karakterleri Kenar Çubuğu dışında) klavye kullanarak Yunan karakterlerine girmenin bir yolunu sunar:

- Bir alfa, <ESC> tuşuna basıp a yazıp ardından <ESC> tuşuna basarak girilir.
- Bir beta, <ESC> tuşuna basıp b yazıp ardından <ESC> tuşuna basarak girilir.
- Bir gama, <ESC> tuşuna basıp c yazıp ardından <ESC> tuşuna basarak girilir.

...ve bunun gibi.

Sistem unicode desteği sağlamazsa, wxMaxima yine de Yunanca karakterleri gösterme yöntemi sağlayacaktır: "Alfa" gibi değişken adlar daima karşılık gelen Yunan simgeleri olarak görüntülenir.

Küçük harfler, hem bir girme harfi girerek veya hem de escape tuşuna basıp ardından gelen Yunan harfinin tüm latin adını girerek girilebilir:

a alpha	i iota	r rho
b beta	k kappa	s sigma
g gamma	l lambda	t tau
d delta	m mu	u upsilon
e epsilon	n nu	f phi
z zeta	x xi	c chi

h eta	om omicron	y psi
q theta	p pi	o omega

Bu durum büyük harfler için de geçerlidir....

A Alpha	I Iota	R Rho
B Beta	K Kappa	S Sigma
G Gamma	L Lambda	T Tau
D Delta	M Mu	U Upsilon
E Epsilon	N Nu	P Phi
Z Zeta	X Xi	C Chi
H Eta	Om Omicron	Y Psi
T Theta	P Pi	O Omega

Bu mekanizma bazı matematiksel sabitler için de çalışır:

hbar	planck's constant: a h with a vertical bar above it
Hbar	a H with a vertical bar above it

Bu mekanizma bazı matematiksel semboller için de çalışır:

2	squared
3	to the power of three
1/2	1/2
partial	partial sign (the d of dx/dt)-Kısmi diferansiyel işareti
integral	integral sign- integral işareti
sq	root
ii	imaginary
ee	element
hb or hbar	h barred
Hbar	H barred
in	in
impl, implies	implies
inf	infinity
empty	empty
TB	Big triangle right

tb	small triangle right
and	and
or	or
xor	xor
nand	nand
nor	nor
equiv	equivalent
not	not
union	union
inter	intersection
subsetq	subset or equal
subset	subset
notsubsetq	not subset or equal
notsubset	not subset
approx	approximately
propto	proportional to
neq, !=, /=	
or #	not equal to
+/- or pm	a plus/minus sign
<= or leq	equal or less than
>= or geq	equal or greater than
<< or ll	much less than
>> or gg	much greater than
equiv	equivalent to
qed	end of proof
sum	sum sign
prod	product sign
exists	"there exists" sign
nexists	"there is no" sign
parallel	a "parallel" sign
perp	a "perpendicular" sign
leadsto	a "leads to" sign
->	a right arrow

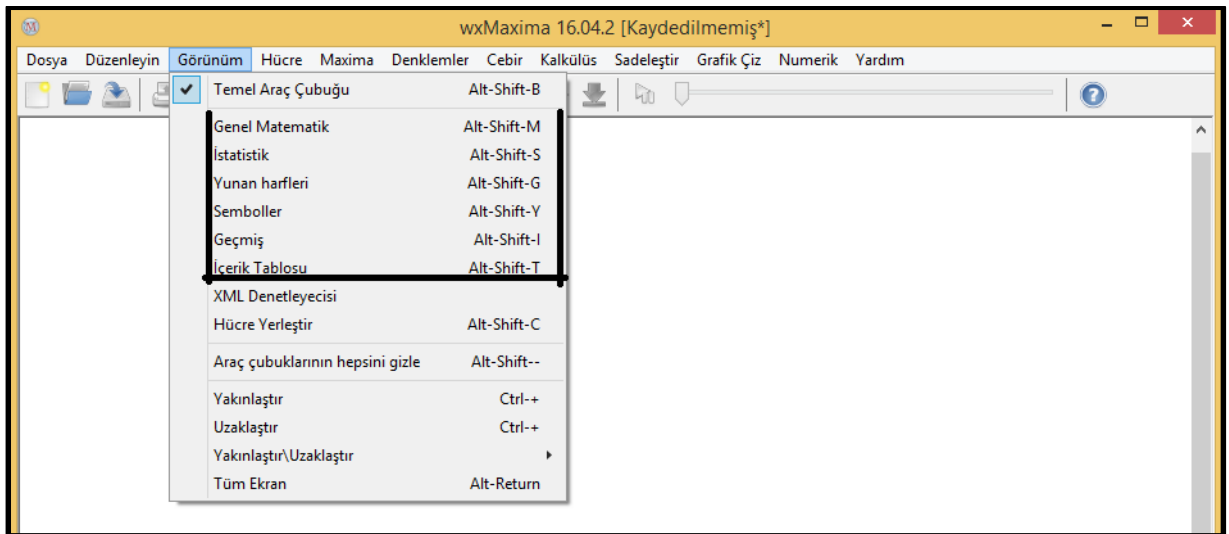
Sembolik formül yazarken yunanca karakterlerin gösterilmesi “%isim” şeklinde de yapılabilir.

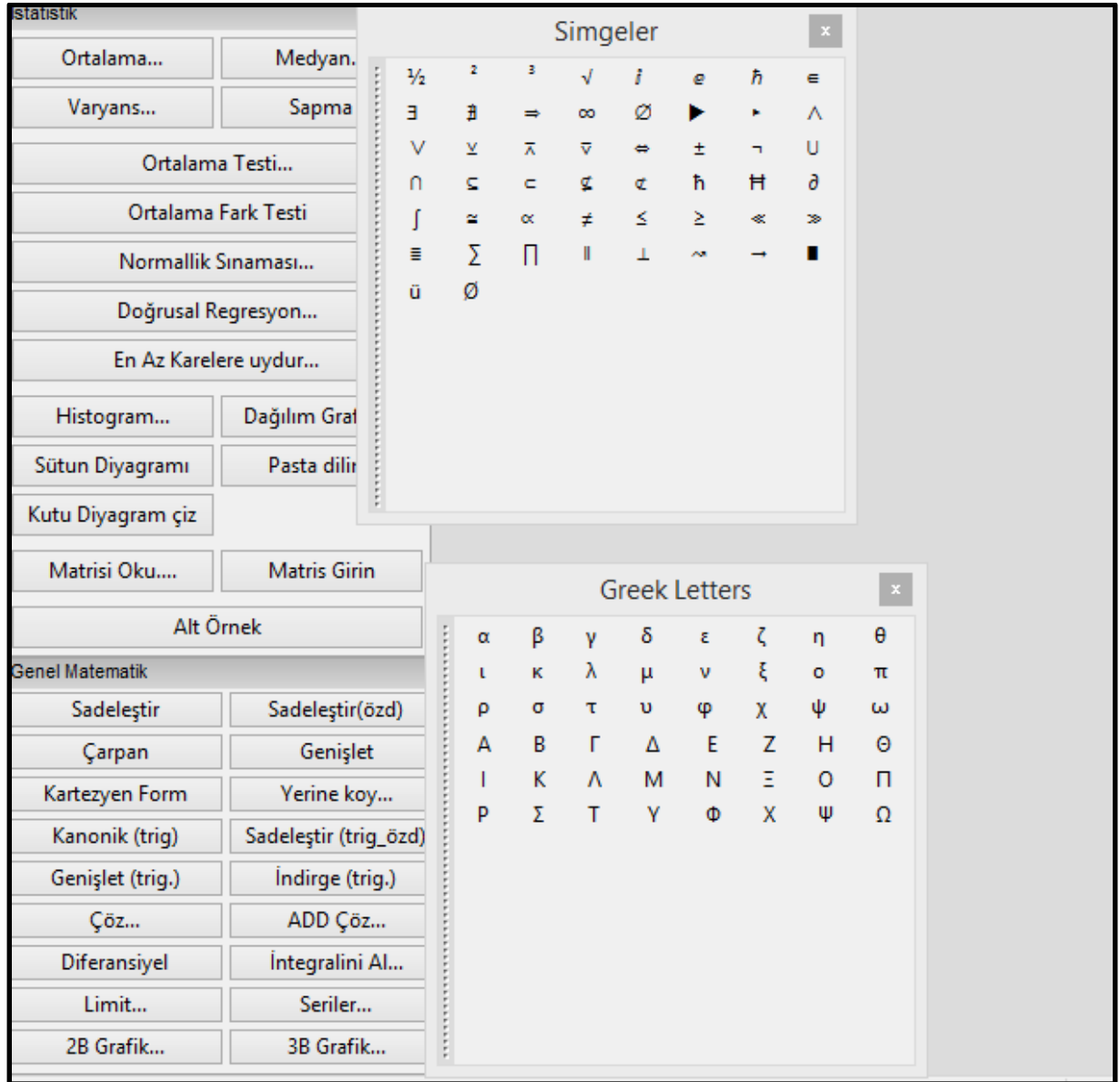
(%i1)	%rho;
(%o1)	ρ
(%i2)	%alpha;
(%o2)	α
(%i3)	%epsilon;
(%o3)	ϵ

Listede özel bir sembol yoksa <ESC> karakterinin numarasına basarak rasgele bir unicode karakteri girmek mümkündür. Bu nedenle, <ESC> 61 <ESC> a ile sonuçlanır. Unicode desteği olmadan lisp'de çalışan bir maksimanın, özel unicode karakterleri içeren dosyalarla uğraşamayabileceğini unutmayın. Bu sembollerin çoğunun (önemli istisnalar mantık sembolleridir), maksimada özel bir anlamı olmadığından, sıradan karakterler olarak yorumlanacaktır. Maksima bir lisp yorumlayıcısı tarafından çalıştırılırsa beklenmedik sonuçlara neden olabilir. Özel unicode karakterlerle desteklemez.

1.2.6. Yan Paneller...Kısayol Tuşları

En önemli Maxima komutlarına yapılan kısayollara ve son çıkan komutların geçmişine yan paneller kullanılarak erişilebilir. "Maxima" menüsündeki "Görünüm" sekmesini tıklayarak girişini kullanarak etkinleştirilebilirler. Kısayollar, konumun wxMaxima penceresinin içinde veya dışında uygun olduğu yere taşınabilir. Aynı komutlara kısayollar üzerinden erişilebilir.





1.2.7. MathML Girdisi

Birkaç kelime işlemci ve benzeri programlar ya MathML girdisini tanıyor ve otomatik olarak düzenlenebilir bir 2D denklem olarak ekliyor - ya da (libreOffice 5.1 gibi) "panodan MathML içe aktar" özelliğini sunan bir denklem düzenleyicisine sahip. Bu soruna cevap vermek için, wxMaxima, bir denklemin seçili bölümünü harici programın (umulur ki) anladığı bir biçimde çıkaran bir "kelime işlemciye kopyala" sağ tıklama menüsü girişi sunar.

1.2.8. Markdown Desteđi

Bir wxMaxima alıřma kitabı, HTML dosyası veya LaTeX dosyası olarak dıřa aktarılabilir. Program, standart markdown kurallarını kullanmaktadır. Birok durumda Markdown, matematik iin sıklıkla kullanılan gsterimlerle arpıřmaktadır. wxMaxima ğeler yıldızlarla iřaretlendiđinde HTML ve TeX dıřa aktarma iin madde imleri listelerini tanımaktadır.

Düz Metin

- * Bir ge, 1. Girinti seviyesi
- * Diđer bir ge. 1.girinti seviyesi
 - * 2.girinti seviyesinde bir ge
 - * 2.girinti seviyesinde ikinci bir ge
- * Birinci girinti seviyesinde 3.bir ge

Düz metin

Markdown ayrıştırıcısının tanıyacađı diđer semboller karşılařtırmalar iin \leq ve \geq , ift-sivri ift ok (\Leftrightarrow), tek bařlı oklar (\leftarrow , \rightarrow ve \leftrightarrow) ve \pm 'yi iřaret. TeX ıkıřı iin \ll ve \gg de tanınır.

1.2.9. Kısayol tuřları

ođu kısayol tuřu, ilgili menülerde bulunabilir. Aslında menü metninden alındıđından ve yerel klavye kullanıcılarının ihtiyalarını karşılamak iin wxMaxima'nın evirileri tarafından zelleřtirilebildiđinden bunları burada belgelemiyoruz. Bununla birlikte, birkaç kısayol tuřu veya kısayol tuř takma adları menülerle belgelenmemiřtir:

- Ctrl + Üst Karakter + Delete, tamamlanmıř bir hücreyi siler.
- Ctrl + Tab veya Ctrl + Shift otomatik tamamlama mekanizmasını tetikler.
- Ctrl + I veya Ctrl + O Yakınlařtırma veya uzaklařtırma.

1.2.10. TeX çıkarımında Raw TeX

Bir metin hücresi TeX: ile başlarsa TeX dışı aktarma işlemi, TeX: işaretçisini izleyen değişmez metin içerir.

1.3. Dosya formatları

Bir wxMaxima oturumunda geliştirilen çalışmalar, daha sonraki kullanımlar için üç türde saklanabilir:

“.mac” dosyası

Makroların kendi kütüphanelerini yazmak için .mac dosyalarını kullanabilirsiniz. Yeterli yapısal bilgi içermediğinden, bir wxMaxima oturumu-belgesi olarak okunamazlar.

“.mac” dosya formatı kendi çalışmalarınız için kullanışlı olan bir dosya türüdür. İsmi kendinizin vereceği Yeni komutlar oluşturabilir ve bir çok alt satırın kalabalığından kurtulabilirsiniz.

“.mac” dosyası yazımı örneği

Örnek :q1 ve q2 elektrik yüklerinde sahip, birbirlerine d uzaklığındaki

iki cismin arasındaki elektirsel kuvveti hesaplayan ($k=9 \cdot 10^9$)

mini bir program yapmak istiyoruz...

```
(%i1) Felektriksel(q1,q2,d):=(f:(q1*q2*9*10^9)/d^2,print("elektriksel kuvvetin  
değeri="),f "N dur.");
```

```
(%i2) Felektriksel(2,3,4);
```

```
Elektriksel kuvvetin değeri= 3375000000 N dur.
```

Kendi formulümüzü oluşturduk.ve çalışıyor olduğunu gördük.

Bu dosyayı "Felektriksel.mac" dosyası olarak

C:\maxima-5.38.1\share\maxima\5.38.1_5_gdf93b7b_dirty\share

dosyasına kaydetmelisiniz. Buraya kaydederseniz yeni bir wxMaxima oturumu açtığınızda komut satırına

load(Felektriksel);

yazdığınızda -kendi mini programınız hazır halde- Felektriksel(q1,q2,d)

komutunu yazmanızı bekleyecektir.

Felektirsel.mac dosyanıza yazmış olduğunuz (varsa başka) komutlarınızı görmek için girdi satırına

values yazıp (Shift+Enter) tıklamanız yeterli olacaktır.

Örnek:Maxima'nın kendi kitaplığında olan **vect.mac** dosyasını

Paket Yükle' den girerek yükleyelim ve hangi işlevlere sahip olduğunu görelim.-----

(%i1) load("C:/maxima-

5.38.1/share/maxima/5.38.1_5_gdf93b7b_dirty/share/vector/vector.mac")\$

(%i2) values (yazıp Shift+Enter' e bas)

[bestlength,trylength,cartesian2d,cartesian3d,polar,polarcylindrical,spherical,elliptic,ellipticcylindrical,confocalelliptic,prolatespheroidalsqrt,oblatespheroidalsqrt,parabolic,paraboliccylindrical,paraboloidal,prolatespheroidal,oblatespheroidal,bipolar,bipolarcylindrical,toroidal,conical,confocalellipsoidal,coords,coordinates,dimension,dimensionbed,sfprod,sf,expandall,expanddot,expanddotplus,expandgrad,expandplus,expandgradplus,expanddiv,expanddivplus,expandcurl,expandcurlplus,expandlaplacian,expandlaplacianplus,expandprod,expandgradprod,expanddivprod,expandcurlcurl,expandlaplacianprod,expandcross,expandcrosscross,expandcrossplus,firstcrossscalar,expandflags,potentialzeroloc,vector,coordvar,scalefactor,coordsystem]

Kullanmayı biraz da olsa öğrenmiş arkadaşlar için güzel bir özellik..Kolay gelsin... ☺
(T.Ş.)

“.wmx” uzantılı dosyalar

.wmx dosyaları maxima için girdi hücrelerinin yanı sıra yazdığınız metin hücreleri, başlık hücreleri ve bölüm veya bölüm hücreleri içerir. Resimlerin ve Maxima'nın çıktısı .wmx dosyasıyla birlikte kaydedilmezler.

“.wxmx” uzantılı dosyalar

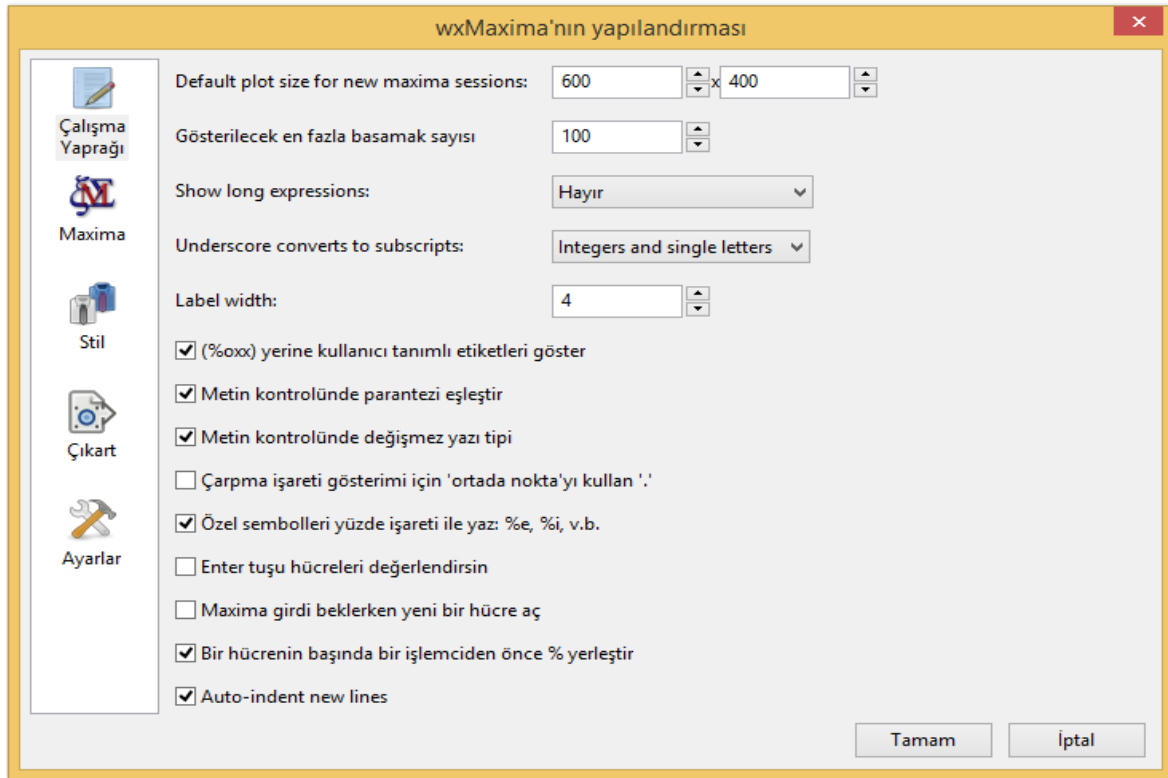
Bu xml tabanlı dosya biçimi çalışma sayfasının içerdiği tüm metinleri ve resimleri kaydeder. Tercih edilen dosya formatıdır ve şu iki avantaja sahiptir:Sürüm denetimi kolay .wxmx biçiminde kaydedilen dosyalar, standart bir sıkıştırılmış biçimde (.png) tüm görüntüleri ve belgenin yapısını ve kullanıcı tarafından yazılan metni içeren xml yapısının sıkıştırılmamış bir kopyasını içerir. Disk alanına-optimize edilmiş .wxmx formatında kaydedilen dosyalar bir bütün olarak sıkıştırılır. Dosyanın saf xml verisi kısmı sıkıştırıldığında temelde daha küçük olma eğilimindedir ve sıkıştırmadan sonra görüntü başlıkları gibi yinelenen veriler, başlangıçta yaptıkları alanın sadece bir kısmını kullanacaktır.Bununla birlikte, sıkıştırılmamış versiyonda tek bir metin satırının değiştirilmesinin sıkıştırılmış bir dosyanın yapısını tamamen değiştirme eğiliminde olmasına karşın maliyetle gelir. Böyle bir dosyayı ele alan bir sürüm kontrol sistemi - bununla birlikte, ikili dosyalar arasındaki farklılıkları işleme konusunda optimize edilmiş olacaktır - bu nedenle, iki dosya sürümü arasında gerekenden çok daha fazla sayıda farklılık izlemelisiniz (ve depolamalıyız); Çoğu sürüm kontrol sistemi sunucudaki sakladığı verileri sıkıştırdığı için hem .wxmx türevleri ilk sürümü tarafından kullanılan sunucu alanı neredeyse aynı olmalıdır.

1.4. Yapılandırma Seçenekleri

WxMaxima bazı yaygın yapılandırma değişkenleri için iki yapılandırma yolunu sunar:

Aşağıdaki yapılandırma iletişim kutusu, geçerli ve sonraki oturumlar için varsayılan değerlerini değiştirmenizi sağlar.

Ayrıca, çoğu yapılandırma değişkeninin değerleri, yalnızca aşağıda gösterildiği gibi çalışma sayfasındaki değerleri geçersiz kılarak geçerli oturum için değiştirilebilir.



1.4.1. Varsayılan animasyon çerçeve hızı

Yeni animasyonlar için kullanılan animasyon çerçeve hızı 'wxanimate_framerate' değişkeninde tutulur . Bu değişkenin yeni bir çalışma sayfasında içerdiği başlangıç değeri, yapılandırma diyalogu kullanılarak değiştirilebilir.

1.4.2. Yeni maxima oturumları için varsayılan grafik boyutu

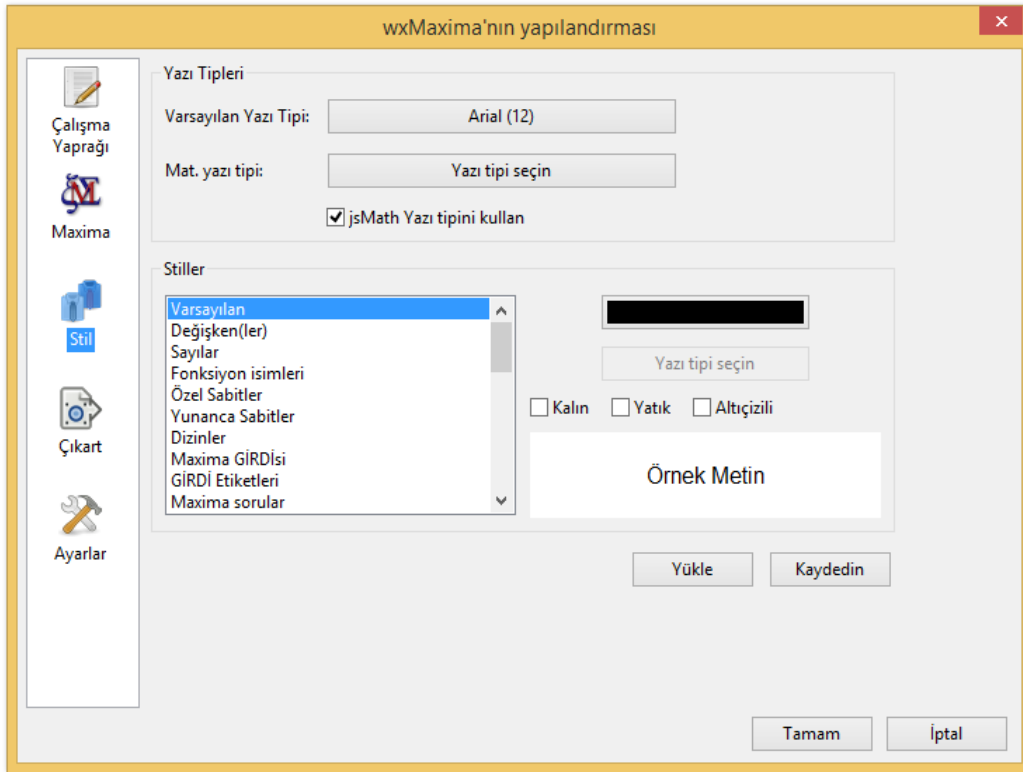
wxplot_size değeri maxima tarafından değiştirilmezse, bir sonraki başlatmanın ardından çalışma sayfasına yerleştirilen çizimler varsayılan boyutlarda oluşturulur.

Tek bir grafiğin çizim boyutunu ayarlamak için,

wxdraw2d(explicit(x^2,x,-5,5)), wxplot_size=[480,480]\$.

1.4.3. “jsMath” fontlarını kullanma

Yapılandırmadaki Stil menüsü, çalışırken dizüstü bilgisayarınızın görünümüyle ilgili özelleştirme sağlar.



Ayrıca, jsMath fontlarını kullanma seçeneği de bulunmaktadır. Özellikle matematik için tasarlandığından ve kırık fontlardan kaynaklanabilecek sorunları çözme eğiliminde oldukları için, bu yazı tiplerini yüklemek iyi bir fikirdir. Bilgisayarınıza yüklenmediyse, etkinleştirmek için seçenek vurgulanmaz. JsMath fontları [bu sitede](#) bulunabilir.

1.4.4. Metin Kontrolünde Eş Parantezler

Bu özellik iki konuda kolaylık sağlar:

- Bir parantez, köşeli parantez veya tırnak açarsanız, wxMaxima otomatik olarak kapanış ekleyecektir.
- Metin seçilirse bu tuşlardan herhangi birine basıldığında seçilen metin, eş işaretlerin arasına yerleştirilir.

1.4.5. Otomatik Kaydetme Aralığı

Bu değer sıfırdan büyük bir değere ayarlanırsa maksima mobil-cihaz benzeri şekilde çalışır:

- Dosyalar çıkışta otomatik olarak kaydedilir. Ve dosya her n dakika sonra otomatik olarak kaydedilir.
- Otomatik kaydetme işlevselliğinin wxMaxima tarafından çalıştırılması için dosyaya bir ad vererek kaydetmiş olmanız gerekir. Bu özellik yalnızca dosya diske önceden kaydedilmiş veya diskten açılmışsa çalışır..

2 Maxima uzantıları

wxMaxima ağırlıklı olarak Maxima için bir grafik kullanıcı arayüzüdür. Yani, esas olarak Maxima'ya komutları iletir ve bu komutların yürütülmesinin sonuçlarını bildirir. Ancak bazı durumlarda, Maxima'ya işlevsellik katar. wxMaxima'nın bir çalışma sayfasının içeriğini HTML ve LaTeX dosyalarına çıkartarak raporlar oluşturma kabiliyeti söz konusudur. Bu bölüm, wxMaxima'nın grafiklerin bir oturuma-çalışma sayfasına eklenmesini geliştirdiği bazı yolları ele alıyor.

2.1. Alt İndisli Değişkenler

wxsubscripts true olarak ayarlanırsa, x_y biçimindeki değişken adları bir alt simge kullanılarak görüntülenir, eğer

- y tek bir harf
- y bir tam sayı ise

Değişken adı bu gereksinimlerle uyuşmuyorsa,

wxdeclare_subscript (variable_name) komutunu kullanarak "alt indisli" olarak bildirilebilir;

veya wxdeclare_subscript ([variable_name1, variable_name2, ...]);

Bir değişkenin alt indisli olarak bildirilmesi aşağıdaki komutu kullanarak geri döndürülebilir:

```
wxdeclare_subscript (variable_name, false);
```

2.2. Grafik Çizme

Grafik çizimi, wxMaxima kullanıcı arayüzünün orijinal programa bazı uzantılar sağlamak zorunda kalacağı bir yerdir.

2.2.1. Çalışma Sayfasına Bir Grafik Çizimi Gömmek

Maxima, harici program “gnuplot”a, oluşturduğu her diyagram için ayrı bir pencere açmasını sağlar. Çoğu zaman, grafikleri çalışma sayfasına yerleştirmek kullanıcı için daha uygundur. wxMaxima, kendi adına kaydedilen karşılık gelen maksima işlevlerden farklı olmayan kendi grafik işlevleri kümesini sağlar: Hepsine bir "wx" öneki eklenir. Örneğin, maxima komutu plot2d'dir. Çizdirilen grafik ayrı bir pencerede açılır. Komut “wxplot2d” şeklinde yazılırsa grafik çizimi çalışma sayfasında gömülü olarak sunulur.”wxdraw” ve”wxhistogram” da maxima’nın draw ve histogram komutlarına karşılık gelir.

2.2.2. Gömülü Çizimleri Büyütme veya Küçültme

Yukarıda belirtildiği gibi, yapılandırma iletişim kutusu, wxplot_size'in başlangıç değerini ayarlayan çizimlerin varsayılan boyutunu değiştirmenin bir yolunu sağlar. WxMaxima'nın çizim yordamları, bir çizimin boyutunu piksel cinsinden belirten bu değişkene duyarlıdır. Her zaman ayarlanabilir. Aşağıdaki örneklerde çizimlerin boyutunun ayarlanması gösterilmiştir:

```
%i1 wxplot_size:[1200,800];
%o1 [1200,800];

%i2 wxdraw2d(
    explicit(
        sin(x),
        x,1,10
    )
) $
```

Tek bir çizimin boyutu değiştirilecekse, maksima, yalnızca geçerli hücre için değiştirme olanağı sağlar.

```
%i1 wxplot_size:[1200,800];
%o1 [1200,800];

%i1 wxdraw2d(
    explicit(
        sin(x),
        x,1,10
    )
),wxplot_size=[1600,800]$

%i1 wxdraw2d(
    explicit(
        sin(x),
        x,1,10
    )
),wxplot_size=[1600,800]$
```

2.2.3. Daha Kaliteli Çizimler

Gnuplot, cairo kütüphanesinin sağladığı yüksek kaliteli bitmap çıktısını destekleyip desteklemediğini belirlemek için bir yol sağlamıyor gibi görünüyor. Gnuplot'un bu kütüphaneyi kullanması için derlendiği sistemlerde yapılandırma menüsündeki pngcairo seçeneği (wxplot_pngcairo değişkeni tarafından geçersiz kılınabilir) antialiasing ve ek çizgi stilleri için destek sağlar.

2.2.4. Çalışma Sayfasına Animasyonlar Yerleştirme

“*with_slider_draw*” komutu, wxdraw2d'nin birden çok çizim hazırlayan ve ekranın üst kısmındaki kaydırıcıyı hareket ettirerek arasında geçiş yapmanıza izin veren bir sürümüdür. ImageMagick yüklüyse, wxMaxima bu animasyonu animasyonlu bir gif olarak dışa aktarmanıza izin verir.

with_slider_draw için ilk iki argüman çizimler arasında basamaklı değişkenin ismi ve bu değişkenlerin değerlerinin bir listesidir. Sonraki argümanlar, wxdraw2d'nin sıradan argümanlarıdır:

```
with_slider_draw(
    f, [1,2,3,4,5,6,7,10],
    title=concat("f=",f,"Hz"),
```

```
explicit(  
    sin(2*%pi*f*x),  
    x,0,1  
) ,grid=true  
);
```

Üç boyutlu çizimler için `with_slider_draw3d` kullanılır.

Kaydırıcıyı kullanan bir başka komut seti de

- `wxanimate_draw`
- `wxanimate_draw3d`:

komutlarıdır.

```
wxanimate_draw(  
    a, 3,  
    explicit(sin(a*x), x, -4, 4),  
    title=printf(false, "a=~a", a));
```

Normalde, animasyonlar wxMaxima'nın yapılandırmasında seçilen kare hızıyla oynatılır veya dışa aktarılır. Hızı ayarlamak için tek bir animasyon oynatılan değişken “`wxanimate_framerate`” kullanılabilir:

```
wxanimate(a, 10,  
    sin(a*x), [x,-5,5]), wxanimate_framerate=6$
```

Animasyon işlevleri, bunları kullanırken farkında olmaları gereken bir tuzağa düşer: Kaydırıcı değişkeninin değeri, çizilecek ifadeyle değiştirilir - değişken ifadede doğrudan görülemezse, başarısız olur. Bu nedenle aşağıdaki örnek başarısız olur:

```
f:sin(a*x);  
with_slider_draw(  
    a,makelist(i/2,i,1,10),  
    title=concat("a=",float(a)),  
    grid=true,  
    explicit(f,x,0,10)  
)$
```

Maxima ifadeyi ilk değerlendirmeye zorlanırsa ve bunun yerine kaydırıcının değerini değiştirmek ister:

```
f:sin(a*x);
  with_slider_draw(
    a,makelist(i/2,i,1,10),
    title=concat("a=",float(a)),
    grid=true,
    explicit('f,x,0,10)
  )$
```

2.2.5. Eşzamanlı Pencereelerde Çoklu Çizimler Açma

WxMaxima tarafından sağlanmamasına rağmen Maxima'nın bu özelliği (bunu destekleyen kurulumlarda) bazen kolayca gelir. Aşağıdaki örnek, Mario Rodriguez'den Maxima posta listesine yazılan bir mesajdan alınmıştır:

```
load(draw);

/* Parabola in window #1 */
draw2d(terminal=[wxt,1],explicit(x^2,x,-1,1));

/* Parabola in window #2 */
draw2d(terminal=[wxt,2],explicit(x^2,x,-1,1));

/* Paraboloid in window #3 */
draw3d(terminal=[wxt,3],explicit(x^2+y^2,x,-1,1,y,-1,1));
```

2.3. Grafikleri Çalışma Sayfasına Gömme

.wxmx dosya biçimi kullanılıyorsa, dosyaları bir wxMaxima projesine gömmek sürükleyip bırak kadar kolayca yapılabilir. Ancak bazen (örneğin, bir resmin içeriği daha sonra bir oturumda değişebilirse), resme değerlendirmeye yüklenecek dosyaya bildirmek daha iyidir: `show_image("Mann.png");`

2.4. “wxmaximarc”

Maxima kullanıcı dizini wxmaxima-init.mac adlı bir metin dosyası içeriyorsa, dosyanın içeriği yeni bir çalışma sayfasının her başlatıldığında otomatik olarak Maxima'ya geçirilir.

Maxima'nın hangi dizini kullanıcı dizini olarak kullandığını bulmak için aşağıdaki satırı yazmanız yeterlidir:

```
maxima_userdir;
```

Maxima'nın cevabı, başlangıç dosyasının yerleştirilebileceği dizinin adını belirtir.

```
%o1 /home/username/.maxima
```

2.5. Özel değişkenler

- `wxsubscripts` maxima'ya bir alt çizgi (`R_150`) veya benzeri içeren değişken adlarını alt indisli değişkenlere dönüştürmesi gerekip gerekmediğini bildirir. Hangi değişken adlarının otomatik olarak dönüştürüldüğü ile ilgili ayrıntılar için `wxdeclare_subscript` sayfasına bakın.
- `wxfilename` Bu değişken, `wxMaxima`'da şu anda açılmış dosyanın adını içerir. Pencerelede bu bilgi sadece yapılandırma diyalogunda Maxima / maxima'nın onay kutusunun belge yolunun kontrolü altındaysa kullanılabilir.
- `wxMaxima`'nın daha fazla çizgi stili ve daha iyi bir grafik kalitesi sağlayan `gnuplot`'un `pngcairo` terminalini kullanmaya çalışıp çalışmadığını `wxplot_pngcairo` bildirir. Bu değişken, yapılandırma iletişim kutusundaki ilgili ayarı okumak veya geçersiz kılmak için kullanılabilir.
- `wxplot_size` gömülü çizimlerin boyutunu tanımlar.
- `wxchangedir` Çoğu işletim sisteminde `wxMaxima`, Maxima'nın çalışma dizinini geçerli dosyanın bulunduğu dizine otomatik olarak ayarlar. Bu, dosya Girdi / Çıktı'nın (örneğin `read_matrix` tarafından) okunması veya yazılması gereken dosyanın tüm yolunu belirtmeden çalışmasına izin verir. Ancak Windows'da bu devre dışı bırakılmıştır: Lisp Standard geçerli çalışma dizini kavramını içermemektedir. Bu nedenle, standart bir ayarlama yolu yoktur ve Maxima'nın yüklü olduğu sürücüde olmayan bir dizine geçilmesi Maxima'nın okumaya başlamasına neden olabilecek bir dizine geçmek için Maxima'nın bulunduğu sürücü yerine kendi sürücüsündeki paket dosyalarını değiştirmeniz gerekir. kurulmuş. `Wxchangedir` ögesinin `true` olarak ayarlanması, `wxMmaxima`'ya bunun riskini ve Maxima'nın çalışma dizini ayarlamasını söyler.

- `wxanimate_framerate` Aşağıdaki animasyonların saniye başına oynatılacak kare sayısı. -1, wxMaxima'ya yapılandırma iletişim kutusundaki varsayılan kare hızını kullanmasını söyler.

2.6. İki Boyutlu Güzel-Yazım : Tablo Yapmak

Komut (`table_form`), Maxima'nın varsayılan çıkış rutininden daha okunaklı biçimde bir 2D liste görüntüler.

```
table_form(  
  [  
    [1,2],  
    [3,4]  
  ]  
)$
```

2.7. Hata Raporlama

wxMaxima mevcut sistem hakkında hata raporlama bilgisi toplayan birkaç işlem sunar:

`wxbuild_info ()`, halen çalışan wxMaxima sürümü hakkında bilgi toplar

`wxbug_report ()` hataların nasıl ve nerede dosyalanacağını bildirir

3 Sorun Giderme

3.1. Maxima'ya Bağlanılamıyor

Maxima (gerçek matematiği yapan program) ve wxMaxima (kullanımı kolay kullanıcı arabirimi), yerel bir ağ bağlantısı vasıtasıyla iletişim kuran ayrı programlardır. Bu nedenle en muhtemel nedeni, bu bağlantı bir şekilde çalışmıyor olmasıdır. Örneğin bir güvenlik duvarı, yalnızca internetten yetkisiz bağlantılara (ve belki de internete bazı bağlantıları kesmekten) engel olmayacak şekilde kurulabilir, ancak güvenlik duvarının içindeki işlemler arası iletişimi de engeller aynı bilgisayarda. Maxima lisp işlemcisi tarafından çalıştırıldığından engellenen işlem iletişiminin mutlaka "maxima" olarak adlandırılması gerekmediğini unutmayın. Ağ bağlantısını açan programın yaygın isimleri `sbcl`, `gcl`, `ccl`, `lisp.exe` veya benzeri isimler olacaktır. Unix bilgisayarlarında, aynı

bilgisayardaki iki program arasında ağ bağlantıları sağlayan geri döngü ağının düzgün yapılandırılmamış olmasının başka bir nedeni olabilir.

3.2. Bozuk bir .wxmx dosyasındaki veriler nasıl saklanır

Dahili olarak en modern xml tabanlı biçimler, bir özel karaktere sahip sıradan zip dosyalarıdır: arşivdeki ilk dosya sıkıştırılmadan depolanır ve bu dosyanın hangi türünün açılacağı hakkında bilgi sağlar.

Çakışmış bir .wxmx dosyasını .zip olarak yeniden adlandırdıktan sonra dosyanın sonundaki zip imzası hala bozulmamışsa, çoğu işletim sistemi içinde depolanan bilgilerin herhangi bir bölümünü ayıklamak için bir yol sağlayacaktır. Bir metin işlemci belgesinden orijinal görüntü dosyalarını kurtarma ihtiyacı olduğunda bu yapılabilir. Zip imzası bozulmamışsa dünyanın sonu olmaz: Kaydetme sırasında wxMaxima bir şeylerin ters gittiğini tespit ettiğinde içerikleri yardımcı olabilecek bir wxmx ~ dosyası olacaktır ve böyle bir dosya olmasa bile : Yapılandırma seçeneği ayarlanmışsa, .wxmx dosyalarının sürüm denetimi için en iyileştirilmesi gerekecektir. Wxmx dosyasını bir .txt dosyasına yeniden adlandırmak ve dosyanın içeriğini kurtarmak için bir metin düzenleyicisi kullanmak mümkündür.

3.3. wxMaxima, Maxima'nın verileri için sonsuza dek bekler

Maxima'nın eski sürümlerinde bunun nedeni, kapanış ayraç, parantez, parantez veya tireleme işaretinin eksik olması olabilir: Bu durumda Maxima, girdisinin geri kalanını alana kadar bekler (asla yapmaz).

Bunun bir sorun olup olmadığını anlamak için, işletim sistemi normalde Maxima'nın gerçekten mevcut sorunu çözmeye çalışarak sonsuza kadar çalışıp çalışmadığını belirlemenin bir yolunu sağlar.

3.4. Komut bitmeden önce ekranda bazı hata ayıklama bilgilerini istiyorum

Normalde wxMaxima, tüm 2D formülün dizgiye başlamadan önce aktarılmasını bekler. Bu, yalnızca kısmen tamamlanmış bir denklem dizerek pek çok girişimde bulunmak için zaman kazandırır. Gerçi hata ayıklama çıktısı verecek bir disp komutu var ve şu anki Maxima komutunun bitmesini beklemiyoruz:

```
for i:1 thru 10 do (  
  disp(i),
```

```
t:makelist(i,i,1000000),  
length(t)  
) $
```

3.5. Windows'daki wxMaxima, basit denklemlerin görüntülenmesi sırasında çöküyor

JsMath yazı tipleri denklemlerin mükemmel 2D görüntüsüne izin verir. Ancak wxMaxima'nın çökmesine neden olan bu paketin kırık sürümleri var. Çalışma sürümü <http://www.math.union.edu/~dpvc/jsmath/download/jsMath-fonts.html> math.union.edu adresinden indirilebilir. WxMaxima'nın aslında bu yazı tiplerini kullanması için wxMaxima'nın yapılandırma diyaloğunun Stiller sekmesinde uygun onay kutusunun etkinleştirilmesi gerekir.

3.6. Animasyonlar çalışmıyor

wxMaxima, animasyonları animasyonlu gif formatına dönüştürmek için bir üçüncü parti araca (ImageMagick) dayanıyor. Bu tabii ki yalnızca bu paket kurulmuş ve sistem tarafından biliniyorsa çalışacaktır (Windows'ta, ImageMagick'in convert.exe dosyasının yolu sistemin bulmasını sağlamak için PATH sistem değişkeninin bir parçası olmalıdır).

Animasyonlu bir gif almamanın bir başka nedeni, bu görüntünün formatın (veya ImageMagick'in) yeteneklerini aşmasıdır:

- .gif dosyaları yalnızca maksimum 256 kareye izin verir
- .gif dosyaları 65535x65535 pikselden fazla olamaz.
- ImageMagick'in tüm kareleri sıkıştırılmamış halde tutması için yeterli belleğe erişmesi gerekir.

3.7. Çizim yalnızca kapalı bir boş zarfı ve hata mesajını gösterir

Bu, wxMaxima'nın Maxima'nın gnuplot'u yaratması için talimat vermesi gereken dosyayı okuyamadığı anlamına geliyor. Bunun neden olabilecek olası sebepler şunlardır: Komplo komutu, implicit_plot gibi üçüncü parti bir paketin bir parçasıdır, ancak bu paket, arsa çizmeye çalışmadan önce Maxima'nın load () komutu tarafından yüklenmemiştir. Maxima, şu anda yüklü olan gnuplot sürümünün anlayamadığı bir şey yapmaya çalıştı. Bu durumda, Maxima'nın maxima_userdir dizini içindeki dizindeki

maxout.gnuplot dosyası, Maxima'nın gnuplot'tan gelen talimatları içerdiğini işaret eder. Bu nedenle, bu dosyanın içeriği, bu nedenle hata ayıklarken yardımcı olur. Gnuplot'a antialiasing ve ilave çizgi stilleri sağlayan pngcairo kütüphanesini kullanma talimatı verildi, ancak bu olasılığı desteklemek için derlenmedi. Çözüm: Yapılandırma iletişim kutusundaki "cairo terminalini çizim için kullan" onay kutusunun işaretini kaldırın ve Maxima'dan wxplot_pngcairo değerini true olarak ayarlamayın. Gnuplot geçerli bir .png dosyası çıkarmadı.

3.8. Bir animasyon çizerken "hata: tanımsız değişken"

Varsayılan olarak kaydırıcı değişkeninin değeri, yalnızca orada görünürse, çizilecek ifadeye değiştirilir. Bu ifade etrafında bir ev () koymak bu sorunu çözmelidir.

3.9. Bir hücre içeriğini kaybettim ve geri alım hatırlamıyor

Hücre işlemleri ve hücrelerdeki değişiklikler için ayrı geri alma işlevleri vardır, bu nedenle şans bu hiç olmadığı kadar düşüktür. Varsa, verileri kurtarmak için birkaç yöntem vardır: wxMaxima aslında iki geri alma özelliğine sahiptir: Herhangi bir hücre seçilmezse etkin olan genel geri alma arabelleği ve imleç bir hücredeyken etkin olan bir hücre başına geri alma arabelleği. Eski bir değere yine de erişilebilir olup olmadığını görmek için geri alma seçeneklerini kullanmaya değer. Maxima'nın hücrenin hangi etiketine atadığı malzemeyi bulmak için hâlâ bir yolunuz varsa, hücrenin etiketini yazmanız yeterlidir ve içerikleri yeniden görünür. Eğer yapmazsan: Paniğe kapılma. "Maxima" menüsünde, son yayınlanan tüm Maxima komutlarını gösteren bir geçmiş paneli göstermenin bir yolu vardır. Başka hiçbir şey yardımcı olmazsa Maxima'nın içerdiği bir tekrarlama özelliği : %i1 playback(); imdadınıza yetişir.

3.10. wxMaxima, "Maxima işlemi Sonlandırıldı" mesajıyla başlar.

Olası bir neden, Maxima'nın wxMaxima'nın yapılandırma iletişim kutusunun "Maxima" sekmesinde ayarlanan konumda bulunamadığı ve bu nedenle hiç çalışmayacağıdır. Bir Maxima ikili dosyasının yolunu belirlemek bu sorunu çözmelidir.

3.11. Maxima sonsuza dek hesaplar ve girdiye cevap vermez

Teorik olarak, wxMaxima'nın Maxima'nın hesaplamayı tamamladığını ve bu yüzden Maxima'ya yeni veri gönderebileceği konusunda asla bilgi sahibi olmadığının farkında

olmadığı anlaşılmaktadır. Durum buysa, "Tetikleyici değerlendirmesi" iki programı yeniden eşitleyebilir.

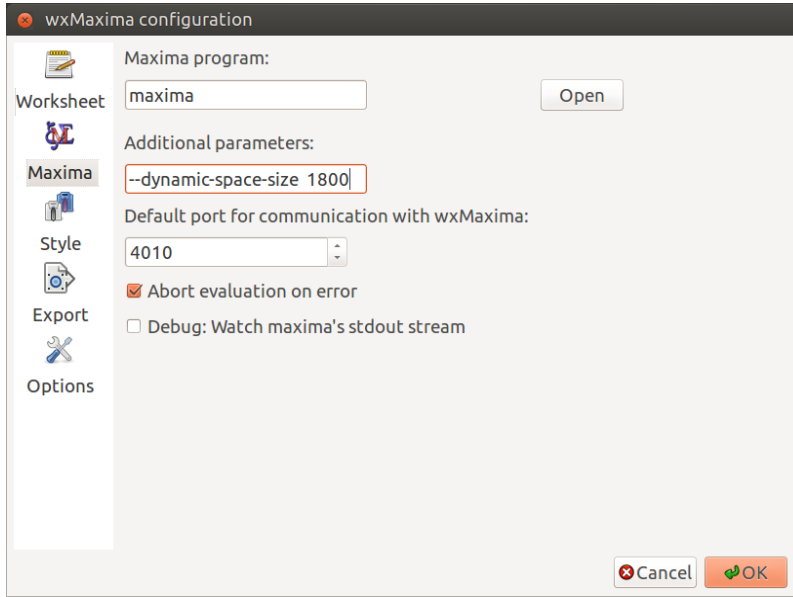
3.12. Maxima'nın Dosya Girdi / Çıktı 'ları Windows'ta çalışmıyor

Windows'ta G / Ç Dosyası varsayılan olarak geçerli dosyanın dizinine göre değil. Maxima dosyasını sürücüde saklarsanız, wxMaxima, wxchangedir ayarına doğru yüklenirse, yük, read_list, batch, read_matrix, save ve benzeri tüm komutları düzeltebilirsiniz. Bu değişkeni true olarak ayarlamak bir dezavantaja sahip olabilir: Maxima hangi dizinin kurulduğunu bilir ve bu dizindeki bir load komutu tarafından talep edilen ek paketi arayacaktır. Ancak yüklü olduğu sürücüyü bilmiyor olabilir. Wxchangedir doğruysa ve geçerli dosya, Maxima'nın Maxima'ya yüklenenden farklı bir sürücüye kaydedildiği için, paketle gönderilen ek paketleri yüklemek başarısız olabilir.

3.13. Benim sbcl tabanlı maxima belleğim tükendi

SBCL varsayılan olarak, düşük uç bilgisayarlarda bile çalışmasına izin veren bir bellek sınırıyla birlikte gelir. Lapack gibi büyük bir yazılım paketini derlerken veya son derece büyük listeler veya denklemler ile uğraşırken bu limit çok düşük olabilir. Sınırları genişletmek için sbcl, sbcl'ye kaç megabayt ayırması gerektiğini bildiren komut satırı parametresi --dynamic-space-size ile sağlanabilir. Bir 32bit-windows-sbcl, 999 Megabayt'a, 1800'e kadar ayırma yapabilir. Windows'da çalışan bir 64 bit sbcl sürümünden, 1280 Megabayt'lık derleme ihtiyaçlarından daha fazlasını kullanması talimatı verilebilir.

Maxima (ve dolayısıyla sbcl) komut satırı parametreleri ile sağlanmanın bir yolu, wxMaxima'nın yapılandırma diyaloğunun ek parametreler alanını oluşturur.



3.14. Giriş bazen durgun / Ubuntu'daki tuşları görmezden geliyor
ibus-gtk Paketini kurmak bu sorunu çözmeli. Ayrıntılar için
(<https://bugs.launchpad.net/ubuntu/+source/wxwidgets3.0/+bug/1421558>) konusuna
bakın.

3.15. Maxima, Yunan karakterlerini veya Umlaut'ları işlediğinde wxMaxima durur
Maximiniz sbcl'ye dayanıyorsa, .sbclrc dosyanıza aşağıdaki satırların eklenmesi
gerekir:

```
(setf sb-impl::*default-external-format* :utf-8)
```

Bu dosyanın yerleştirilmesi gereken klasör, sisteme ve kuruluma özgüdür. Ancak şu
anki oturumda bir hücreyi değerlendirmiş olan herhangi bir sbc tabanlı Maxima,
aşağıdaki komutu aldıktan sonra onu mutlaka bulacaktır:

```
:lisp (sb-impl::userinit-pathname)
```

3.16. Grafik Çizdirme

3.16.1 WxMaxima'nın hem görüntü dosyalarını hem de gömülü grafikleri bir
kerede çıkartabilmesini sağlayabilir miyim?

Aşağıdaki iki işlevin yaptığı gibi resimleri ilk önce üretmek ve daha sonra tekrar içe
aktarmak daha kolaydır:

```

load("draw");
pngdraw(name,[contents]):=
(
    draw(
        append(
            [
                terminal=pngcairo,
                dimensions=wxplot_size,
                file_name=name
            ],
            contents
        )
    ),
    show_image(sprintf(false,"~a.png",name))
);
pngdraw2d(name,[contents]):=
    pngdraw(name,gr2d(contents));

pngdraw2d("Test",
    explicit(sin(x),x,1,10)
);

```

3.16.2. Bir çizim ebatlarını ayarlayabilir miyim?

Bunu Maxima'yı doğrudan kullanarak yapamazsınız. Ancak bunun için gnuplot komutları var:

```

wxdraw2d(
    user_preamble="set size ratio 1;set tmargin 3;set bmargin
3;set lmargin 3;set rmargin 3",
    explicit(sin(x),x,1,10)
),wxplot_size=[1000,1000];

```

4 SSS-Sıkça Sorulan Sorular

4.1. Bir pdfLaTeX sayfasında daha fazla metin uydurmanın bir yolu var mı?

Var: Sadece LaTeX giriş bölümüne aşağıdaki satırları ekleyin (örneğin, yapılandırma iletişim kutusundaki ilgili alanı kullanarak):

```

\usepackage[a4paper,landscape,left=1cm,right=1cm,top=1cm,bottom=1cm]{geometr
y}

```

5 Komut satırı bağımsız değişkenleri

Çoğu işletim sistemi komut satırından daha az karmaşık program başlatma yolları sunar, bu nedenle bu olasılık yalnızca nadiren kullanılır. wxMaxima yine de bazı komut satırı anahtarları sağlar.

-v veya --version: Sürüm bilgilerini çıkartır

-h veya --help: Kısa bir yardım metni çıktılar

-o veya --open: Bu komut satırı anahtarına argüman olarak verilen dosya adını açın

-b veya --batch: Komut satırı bir dosyayı açarsa, bu dosyadaki tüm hücreler değerlendirilir ve daha sonra dosyalar kaydedilir. Bu, örneğin dosyada açıklanan oturum Maxima'nın çıktı dosyalarını üretmesi durumunda yararlıdır. WxMaxima Maxima'nın bir hata çıkardığını algılar ve Maxima'nın bir sorusu varsa duraklatırsa, toplu işleme durdurulur: Matematik, etkileşimli tamamen etkileşimli bir toplu işleme her zaman garanti edilemeyeceği için doğası gereği interaktiftir.

(Yalnızca pencerelerde): -f veya --ini: Bu komut satırı anahtarına argüman olarak verilen init dosyasını kullanın

Bir eksi yerine, bazı işletim sistemleri komut satırı anahtarlarının önüne çizgi kullanabilir.

Kısa İçerik

[WxMaxima kullanıcı kılavuzu](#)

[1 wxMaxima'ya Giriş](#)

[2 Maksima uzantıları](#)

[3 Sorun Giderme](#)

[4 SSS](#)

[5 Komut satırı bağımsız değişkeni](#)

6 Table of Contents

1	wxMaxima'ya Giriş.....	1
1.1.	Maxima ve wxMaxima	1
1.1.1.	Maxima.....	2
1.1.2.	wxMaxima	3
1.2.	wxMaxima Çalışma Sayfası	4

1.2.1.	wxMaxima Çalışma Sayfası Yaklaşımı	4
1.2.2.	Hücreler	6
1.2.3.	Yatay ve Dikey İmleçler	6
1.2.4.	Komutların Otomatik Olarak Tamamlanması:	7
1.2.5.	Yunanca karakterler	8
1.2.6.	Yan Paneller...Kısayol Tuşları	11
1.2.7.	MathML Girdisi	12
1.2.8.	Markdown Desteği	13
1.2.9.	Kısayol tuşları	13
1.2.10.	TeX çıkarmında Raw TeX.....	14
1.3.	Dosya formatları	14
	“.mac” dosyası	14
	“.wxm” uzantılı dosyalar	15
	“.wxmx” uzantılı dosyalar	16
1.4.	Yapılandırma Seçenekleri	16
1.4.1.	Varsayılan animasyon çerçeve hızı	17
1.4.2.	Yeni maxima oturumları için varsayılan grafik boyutu	17
1.4.3.	“jsMath” fontlarını kullanma.....	18
1.4.4.	Metin Kontrolünde Eş Parantezler	18
1.4.5.	Otomatik Kaydetme Aralığı	19
2	Maxima uzantıları	19
2.1.	Alt İndisli Değişkenler	19
2.2.	Grafik Çizme	20
2.2.1.	Çalışma Sayfasına Bir Grafik Çizimi Gömme.....	20
2.2.2.	Gömülü Çizimleri Büyütme veya Küçültme	20
2.2.3.	Daha Kaliteli Çizimler	21
2.2.4.	Çalışma Sayfasına Animasyonlar Yerleştirme	21
2.2.5.	Eşzamanlı Pencereelerde Çoklu Çizimler Açma	23
2.3.	Grafikleri Çalışma Sayfasına Gömme	23
2.4.	“wxmaximarc”	23
2.5.	Özel değişkenler	24
2.6.	İki Boyutlu Güzel-Yazım : Tablo Yapmak	25
2.7.	Hata Raporlama	25
3	Sorun Giderme	25
3.1.	Maxima'ya Bağlanılamıyor.....	25

3.2.	Bozuk bir .wxmx dosyasındaki veriler nasıl saklanır	26
3.3.	wxMaxima, Maxima'nın verileri için sonsuza dek bekler	26
3.4.	Komut bitmeden önce ekranda bazı hata ayıklama bilgilerini istiyorum.....	26
3.5.	Windows'daki wxMaxima, basit denklemlerin görüntülenmesi sırasında çöküyor	27
3.6.	Animasyonlar çalışmıyor	27
3.7.	Çizim yalnızca kapalı bir boş zarfı ve hata mesajını gösterir	27
3.8.	Bir animasyon çizerken "hata: tanımsız değişken"	28
3.9.	Bir hücre içeriğini kaybettim ve geri alım hatırlamıyor	28
3.10.	wxMaxima, "Maxima işlemi Sonlandırıldı" mesajıyla başlar.	28
3.11.	Maxima sonsuza dek hesaplar ve girdiye cevap vermez	28
3.12.	Maxima'nın Dosya Girdi / Çıktı 'ları Windows'ta çalışmıyor	29
3.13.	Benim sbcl tabanlı maxima belleğim tükendi.....	29
3.14.	Giriş bazen durgun / Ubuntu'daki tuşları görmezden geliyor	30
3.15.	Maxima, Yunan karakterlerini veya Umlaut'ları işlediğinde wxMaxima durur 30	
3.16.	Grafik Çizdirme	30
3.16.1	WxMaxima'nın hem görüntü dosyalarını hem de gömülü grafikleri bir kerede çıkartabilmesini sağlayabilir miyim?.....	30
3.16.2.	Bir çizim ebatlarını ayarlayabilir miyim?.....	31
4	SSS-Sıkça Sorulan Sorular	31
4.1.	Bir pdfLaTeX sayfasında daha fazla metin uydurmanın bir yolu var mı?	31
5	Komut satırı bağımsız değişkenleri.....	32

