

بسمه می تعالی

نرم افزار ریاضی و آمار

استاد حجت قجر

دانشکده فنی و حرفه‌ای انقلاب اسلامی

۱۳۹۴

طراح: حسام غلامی

## فهرست مطالب

نرم افزار Maple:	۴
وارد کردن دستورات:	۴
عملگرهای نرم افزار Maple:	۵
تابع Function Advisor:	۶
امکانات بیشتر بوسیله ی منوی راست کلیک:	۶
فراخوانی توابع موجود در بسته ها و زیر بسته ها:	۶
بسته ی RealDomain:	۷
تابع solve:	۸
تابع expand:	۸
تابع limit:	۹
تابع surd:	۹
ریاضیات پایه در Maple:	۹
تعریف تابع:	۹
ترکیب توابع:	۱۱
چند تابع مهم استاندارد در Maple:	۱۲
floor:	۱۲
ceil:	۱۲
round:	۱۲
trunc:	۱۳
frac:	۱۳
rand:	۱۳
معرفی برخی توابع موجود در زیر بسته های بسته ی Student:	۱۴
معرفی برخی از توابع موجود در زیر بسته ی Precalculus:	۱۴

۱۴.....	تابع PolynomialTutor
۱۴.....	تابع RationalFunctionTutor
۱۵.....	تابع StandardFunctionTutor
۱۵.....	تابع ConicsTutor
۱۵.....	تابع CompleteSquare
۱۶.....	تابع مرکز جرم CenterOfMass
۱۶.....	تابع Line
۱۷.....	تابع Slope
۱۷.....	تابع Distance
۱۷.....	بررسی برخی توابع موجود در زیر بسته‌ی Calculus1
۱۷.....	تابع AntiderivativeTutor
۱۷.....	تابع AntiderivativePlot
۱۸.....	تابع DerivativeTutor
۱۸.....	تابع DiffTutor
۱۸.....	تابع IntTutor
	قضیه‌ی مقدار میانگین و محاسبه‌ی آن در Maple به کمک تابع
۱۸.....	MeanValueTheoremTutor
۱۹.....	تابع ArcLengthTutor

# نرم افزار Maple:

نرم افزار Maple یکی از توانمند ترین برنامه های قابل استفاده است که در زمینه های مختلف ریاضی به کار میرود. این نرم افزار برای هریک از مفاهیم ریاضی از جمله اعداد، چند جمله ای ها، متغیر ها و توابع، حد، مشتق، انتگرال، معادلات دیفرانسیل، جبر خطی، دنباله ها، سری ها، تحقیق در عملیات، هندسه، آمار و ... توابع پیش ساخته ای دارد. بیشتر کارها در Maple توسط توابع پیش ساخته انجام میگردد که در صف بسته یا زیر بسته قرار گرفته اند. کل توابع پیش ساخته تقریباً ۳۳۰۰ عدد میباشند. علاوه بر این، امکان تعریف تابع یا توابع جدید نیز وجود دارد. همچنین Maple ابزاری در اختیار دارد که به کمک آنها میتوان توابع پیش ساخته با برخی از زبان های کامپیوتری را فراخوانی کند.

اکثر توابع که در بسته های گوناگون قرار دارند، تنها با فعال کردن آن بسته ها قابل اجرا هستند. فعال کردن یک بسته با دستور **with** انجام میگردد.  
مثلاً دستور مقابل:

```
with(plots);
```

بسته ی **plots** را فعال میکند و بعد از این دستور میتوان از تمامی توابع آن استفاده کرد. برای خروج از بسته میتوان از دستور **restart** استفاده کرد. برخی از توابع Maple بدون هیچ دستوری نیز قابل استفاده هستند که ما آنها را توابع استاندارد (Standard) و اولیه (Initially) مینامیم.

## وارد کردن دستورات:

در سمت چپ هر صفحه از کار برگ، مستطیلی وجود دارد که نشان دهنده ی محل وارد کردن دستور یا دستورات است. هر دستور میتواند به یکی از کاراکتر های **:** و **;** منتهی شود. دستوری که به کاراکتر **:** ختم شود خروجی آن نمایش داده نمیشود ولی اجرا میگردد. مثلاً:

```
Ln(6):
```

```
Ln(5);
```

`2^10;`

در یک سطر میتوان یک یا چند دستور را وارد کرد. همچنین یک دستور را میتوان در یک یا چند سطر وارد کرد. در حالتی که یک دستور طولانی است و مجبورید بخشی از آن را در سطر بعدی بنویسید، لازم است از کلید ترکیبی *Shift + Enter* استفاده کنید.

برای آنکه گروهی از دستورات را با هم اجرا کنید، میتوانید از همین دستور نیز استفاده کنید. برای حذف یک دستور با ماوس روی دستور مورد نظر کلیک کرده و سپس کلید ترکیبی *Ctrl + Delete* را فشار دهید.

---

توجه: عدد  $\pi$  با علامت  $Pi$  نوشته میشود.

---

مثال:

`Sin(Pi/6);`

## عملگرهای نرم افزار Maple:

نرم افزار Maple انواع مختلفی از عملگرها را در اختیار کاربر قرار میدهد که به معرفی برخی از آنها میپردازیم.

<code>&lt;=</code>	کوچکتر یا مساوی	<code>+</code>	جمع
<code>&gt;</code>	بزرگتر	<code>-</code>	تفریق
<code>&gt;=</code>	بزرگتر یا مساوی	<code>*</code>	ضرب
<code>=</code>	مساوی	<code>/</code>	تقسیم
<code>&lt;&gt;</code>	نا مساوی	<code>^</code>	توان
<code>-&gt;</code>	تفریق توابع	<code>&lt;</code>	کوچکتر
<code>:=</code>	انتساب	<code>@</code>	ترکیب
<code>%</code>	عبارت آخری	<code>\$</code>	دنباله

سومین عبارت از آخر %%%

دومین عبارت از آخر %%

## تابع Function Advisor:

تابعی استاندارد است که میتواند اطلاعات مفیدی را درباره‌ی برخی مفاهیم ریاضی و توابع ریاضی به کاربر ارائه دهد.

مثال:

```
FunctionAdvisor(ln);  
FunctionAdvisor(definition, ln);
```

### امکانات بیشتر بوسیله‌ی منوی راست کلیک:

در نرم افزار Maple بوسیله‌ی منوی راست کلیک میتوان امکانات بیشتری را در اختیار داشت. برای مثال، برای نمایش ارقام بیشتر در هنگام محاسبات، روی پاسخ راست کلیک کرده و از منوی متن باز شده گزینه‌ی Approximate را انتخاب کنید.

همچنین امکانات دیگر مانند رسم نمودار بوسیله‌ی گزینه‌ی Plot، گرفتن مشتق بوسیله‌ی گزینه‌ی Differentiate و ... در دسترس میباشد.

### فراخوانی توابع موجود در بسته‌ها و زیر بسته‌ها:

فراخوانی توابع در Maple به روش‌های مختلفی انجام میگردد. دو فرم متداول برای فراخوانی توابع را معرفی میکنیم.

توابع داخلی بیشتر بسته‌ها را میتوان به شکل کوتاه و یا بلند، فراخوانی کرد که عبارت است از:  
فرم کوتاه:

```
with(Package);  
Function(Parameters);
```

فرم بلند:

```
Package[Function](Parameters)
```

مثلاً:

```
A := Matrix([[ -13, -10], [21, 16]]):
```

فرم کوتاه:

```
with(LinearAlgebra):  
MatrixPower(A,2);
```

فرم بلند:

```
LinearAlgebra[MatrixPower](A,2);
```

همچنین برای فراخوانی یک زیر بسته میتوان از دو فرم کوتاه و بلند استفاده کرد.

فرم کوتاه:

```
with(Package[Subpackage]):  
Function(Parameters);
```

فرم بلند:

```
Package[Subpackage][Function](Parameters);
```

## بسته‌ی RealDomain:

```
with(RealDomain);
```

این بسته شامل ۳۸ تابع است. حوزه‌ی کار توابع این بسته، مجموعه‌ی اعداد حقیقی می‌باشد. برخی از توابع این بسته را بررسی می‌کنیم.

## تابع solve:

به کمک این تابع می‌توان معادلات و نا معادلات جبری را حل کرد. به مثال های زیر توجه کنید:

```
solve(4x - 2y = 8, y);  
solve(x^2 + y^2 = 1, x);  
solve(x^2 - x = 2015, x);  
solve({2x - 3y = 7, 3x + 2y = 11}, {x, y});  
solve({x + y + z = 7, x + y - z = 5, x - y + z = 3}, {x,  
y, z});
```

به کمک **solve** نا معادلات را نیز می‌توان حل کرد.

مثال:

```
solve({x + y < 10, x^2 = 9}, {x, y});
```

## تابع expand:

یکی دیگر از توابع این بسته، تابع **expand** (بست دادن، باز کردن) می‌باشد.

مثال:

```
expand((x + 3) * (x + 7));  
expand((x + 1) / (x + 3));  
expand(sin(a + b));  
expand(sin(2x));
```



## تابع `limit`:

تابع بعدی تابع `limit` میباشد که برای محاسبه‌ی حد توابع مورد استفاده قرار میگیرد.

مثال:

```
limit(sin(x) / x, x = 0);  
limit((x^2 - 7x + 6) / (x^2 - 1), x = 1);
```

## تابع `surd`:

تابع بعدی تابع `surd` میباشد که برای ریشه گرفتن از اعداد به کار میرود.

مثال:

```
surd(x, y);  
surd(x^9, 3);
```

# ریاضیات پایه در Maple:

## تعریف تابع:

اگر ضابطه‌ی یک تابع را داشته باشیم، برای معرفی آن در Maple از دستور زیر استفاده میکنیم:

```
f := x -> ضابطه;
```

مثال:

```
f := x -> x^2 + sin(x);  
f(0);  
f(Pi);
```

میتوان توابع را دو ضابطه ای معرفی کرد:

```
g := (x, y) -> ضابطه;
```

مثال:

```
g(x, y) = (x - y) / (x + y);  
g(1, 2);
```

میتوان توابع برداری را نیز معرفی کرد:

```
h := t -> ( x(t), y(t) );
```

یا

```
h := t -> ( x(t), y(t), z(t) );
```

مثال:

```
h := t -> ( t, t^2, 1/t );  
k := s -> ( ln(s), exp(-s) );
```

---

نکته: تابع `exp` یکی از توابع *Maple* است که عدد نپیر را به توان ورودی میرساند.

---

مثال:

```
exp(x);
```

---

تذکر: تابع `evalf` تابعی استاندارد است که برای ارزیابی عبارت های عددی پیچیده استفاده میشود.

---

مثال:

```
evalf[5](Pi);
```

قطعه کد بالا عدد  $\pi$  را تا ۵ رقم نمایش میدهد.

### ترکیب توابع:

با استفاده از عملگر @ میتوان توابع ریاضی را با هم ترکیب کرد.

مثال:

```
f := x -> 1/x;  
g := x -> sin(x);  
(g@f)(x);
```

خروجی قطعه کد بالا  $\sin(1/x)$  خواهد بود.

اگر بخواهیم یک عبارت را نام گذاری کنیم، از دستور زیر استفاده میکنیم:

```
q := ضابطه;
```

مثال:

```
q := 1 + cos(x);
```

تابع `unapply` تابعی استاندارد است که برای تبدیل یک عبارت به تابع مورد استفاده قرار میگیرد:

```
r := unapply( q, x);
```

قطعه کد بالا از عبارت `q` یک تابع بر حسب `x` میسازد.

```
r(1);
```

قطعه کد بالا نیز تابع ایجاد شده را با مقدار ۱ اجرا میکند.

# چند تابع مهم استاندارد در Maple :

## :floor

تابعی استاندارد است، این تابع یک عدد را دریافت کرده و عدد صحیح کوچکتر یا مساوی آن عدد را بر می گرداند. (تابع جزء صحیح)

مثال:

```
floor(-3.718);  
floor( -exp(3) );  
floor( tan(1) );
```

## :ceil

تابعی استاندارد است که عدد اعشاری را دریافت کرده و عدد صحیح بزرگتر و یا مساوی با آن را بر می گرداند.

مثال:

```
ceil( -exp(3) );
```

## :round

تابعی استاندارد است که عدد را دریافت کرده و آن را به نزدیک ترین عدد صحیح گرد میکند.

مثال:

```
round(7.81);  
round(7.49);  
round(7.5);
```

```
round(-7.5);
```

## :trunk

تابعی استاندارد است که عددی اعشاری را دریافت کرده و قسمت اعشاری را حذف کرده و عدد صحیح آن را پس میدهد.

مثال:

```
trunk( $\sqrt{257}$ );
```

## :frac

تابعی استاندارد است که عددی اعشاری را دریافت کرده و قسمت اعشاری یا کسری آن را بر میگرداند.

مثال:

```
frac(8/3);
```

## :rand

تابعی استاندارد است که برای تولید اعداد تصادفی به کار میرود:

```
rand();
```

خروجی حاصل از این تابع همواره یک عدد صحیح مثبت ۱۲ رقمی است. اما این خروجی میتواند توسط کاربر نیز تعیین شود.

برای مثال، دستور زیر دو عدد تصادفی بین صفر و ۲۰ تولید میکند:

```
r := rand(20): r(); r();
```

همچنین میتوان محدوده‌ی اعداد را نیز مشخص کرد. برای مثال دستور زیر دو عدد تصادفی بین اعداد ۲۰ و ۹۴ تولید خواهد کرد:

```
r := rand( 20..94 ): r(); r();
```

# معرفی برخی توابع موجود در زیر بسته های بسته‌ی Student:

```
with(Student);
```

## معرفی برخی از توابع موجود در زیر بسته‌ی Precalculus:

Student دارای زیر بسته‌ای است با عنوان Precalculus که تابع های آن را بررسی میکنیم:

```
with(Precalculus);
```

---

نکته: توابعی که انتهای آنها عبارت Tutor قرار دارد، آموزشی است.

---

## تابع PolynomialTutor:

برای آشنایی بیشتر با نمودار توابع چند جمله‌ای مورد استفاده قرار می‌گیرد.

مثال:

$$3 * x^3 + 2 * x^2 - x$$

## تابع RationalFunctionTutor:

برای آموزش و رسم توابع گویا مورد استفاده قرار می‌گیرد.

مثال:

$$y=(x^2 + 1)/ (x + 1)$$

## تابع StandardFunctionTutor:

شامل همه‌ی تابع های استاندارد است که تا به حال داشته ایم. مانند:  $\sin$ ,  $\cos$  و ...

به کمک آنها میتوان نمودار توابعی که با انتقال، انقباض و یا انبساط تغییر می کنند را رسم کرد.

مثال:

$$3*\sin(2*x) + 1$$

$$1 - 2*\exp(3*x - 1)$$

## تابع ConicsTutor:

شامل آموزش مقاطع مخروطی شامل بیضی، دایره، سهمی، هذلولی و ... میباشد.

مثال:

$$x^2 - 3*y^2 + x + y + 3 = 0$$

$$x^2 - y^2 = 1$$

## تابع CompleteSquare:

این تابع یک عبارت را گرفته و مربع کامل آن را برمیگرداند.

مثال:

$$\text{CompleteSquare}(x^2 - 4x + 7);$$

$$\text{CompleteSquare}(x^2 - 4x + y^2 + 8y + 1);$$

$$\text{CompleteSquare}\left(\int \left( (x+1) / (x^2 + 4x) dx \right) \right);$$

---

نکته: برای درج علامت  $\int$  از ابزارهای موجود در سمت چپ نرم افزار *Maple* از قسمت Expression/استفاده میکنیم.

---

اگر بعد از عبارت  $x$  یا  $y$  بگذاریم *Maple* آن عبارت را برحسب  $x$  یا  $y$  بدست می آورد.

مثال:

```
CompleteSquare(x^2 - 4x + y^2 + 8y + 1, x);
```

### تابع مرکز جرم CenterOfMass:

این تابع برای پیدا کردن مختصات مرکز جرم چند نقطه در صفحه می باشد. ورودی این تابع باید حداقل ۲ نقطه باشد.

مثال:

```
CenterOfMass([-1, 3] , [7, -5]);
```

### تابع Line:

با کمک این تابع می توانیم معادله ی خطی را که شیب و عرض از مبدا آن مشخص است را بدست بیاوریم. در این تابع اولین ورودی شیب خط و دومین ورودی عرض از مبدا میباشد.

مثال:

```
Line(-2, 3);
```

اگر بخواهیم که جواب به صورت یکجا و به صورت شکل نمایش داده شود مانند دستور زیر عمل میکنیم:

```
Line(-2, 3, output=plot);
```



## تابع Slope:

این تابع برای محاسبه‌ی شیب خط بین دو نقطه و یا شیب خط واصل بین دو نقطه روی یک منحنی به کار می‌رود.

مثال:

```
Slope([3, -4], [1, 1]);  
Slope(3x - 4);
```

## تابع Distance:

این تابع فاصله‌ی بین دو نقطه را بدست می‌آورد. نقاط می‌توانند روی یک محور یا در فضای دو بُعدی و یا در فضای سه بُعدی باشند.

مثال:

```
Distance([-1, 1], [3, 7]);
```

## بررسی برخی توابع موجود در زیر بسته‌ی Calculus1:

```
with(Calculus1);
```

## تابع AntiderivativeTutor:

این تابع برای آموزش انتگرال می‌باشد.

## تابع AntiderivativePlot:

این تابع برای رسم نمودار انتگرال می‌باشد.

## تابع DerivativeTutor:

این تابع برای آموزش مشتق گیری میباشد.

## تابع DiffTutor:

این تابع برای آموزش گام به گام دیفرانسیل میباشد.

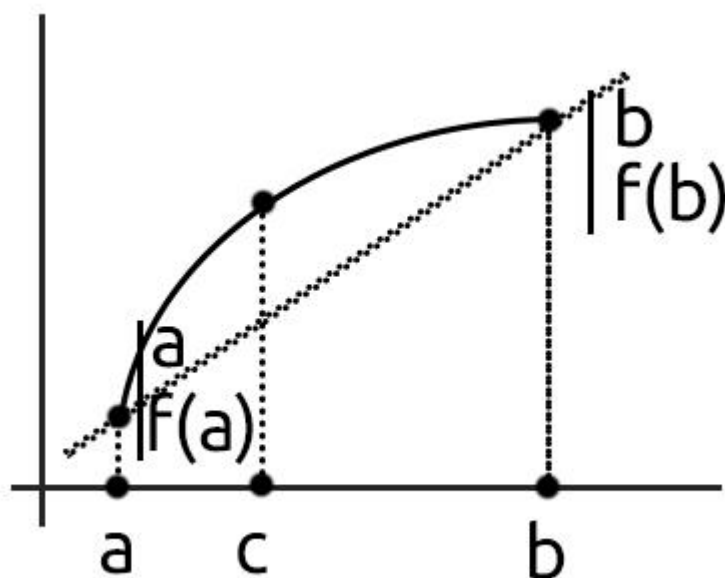
## تابع IntTutor:

این تابع برای آموزش گام به گام انتگرال گیری میباشد.

## قضیه‌ی مقدار میانگین و محاسبه‌ی آن در Maple به کمک تابع

### :MeanValueTheoremTutor

این قضیه بیان میکند که روی منحنی یک تابع خطی مماس میتوان رسم کرد که موازی با پاره خط واصل بین ابتدا و انتهای منحنی است. شیب پاره خط واصل مقدار متوسط تابع را نشان میدهد.



مقدار متوسط شکل بالا از رابطه‌ی زیر به دست می‌آید:

---

$$\frac{f(b) - f(a)}{b - a} = f'(c)$$

---

در Maple برای بدست آوردن مقدار متوسط از تابع `MeanValueTheoremTutor` استفاده میشود.

### تابع `ArcLengthTutor`:

به کمک این تابع میتوان طول یک منحنی را در یک فاصله‌ی داده شده بدست آورد.