Procedimentos Graficos em Calculo Integral

Universidade Estadual de Maringá
Departamento de Matemática
Prof. Doherty Andrade (DMA- UEM)
Prof. Timothy M. (WLU-USA)

Maple

Procedimentos para região de integral dupla em coordenadas retangulares que retorna uma estrutura de plot. Estes procedimentos tomam duas variaveis como argumentos. Execute a worksheet e faça os exemplos.

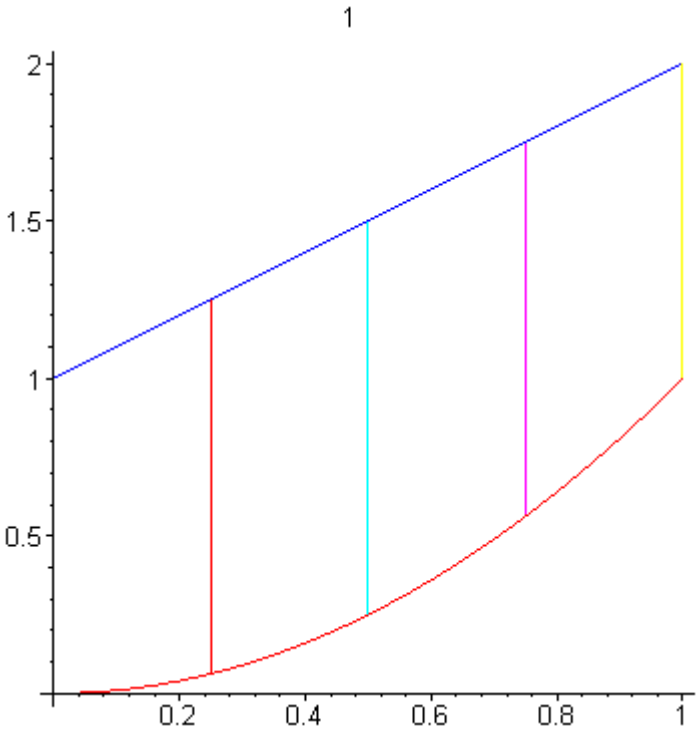
Execute o procedimento e faça os exemplos.

O Procedimento (execute-o)

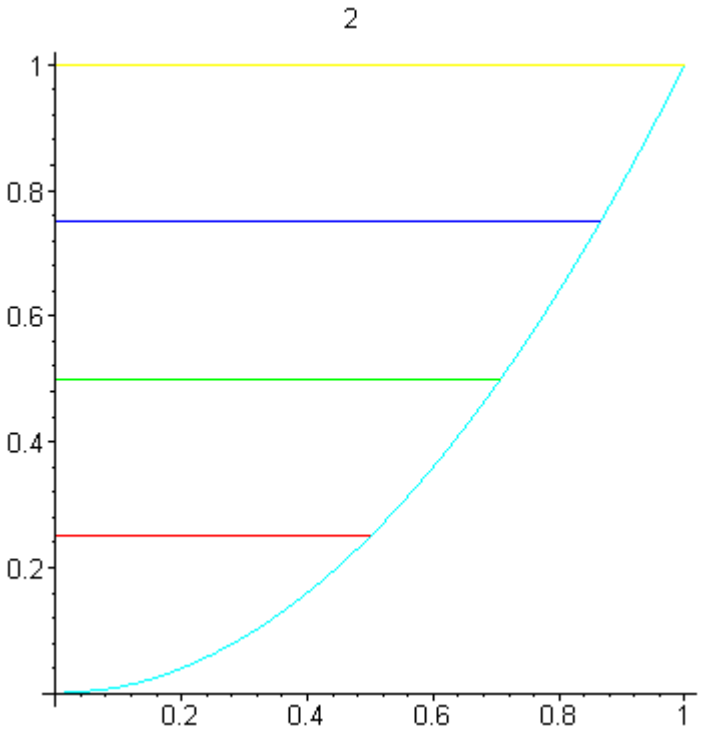
Exemplos

Exemplos de dydxplot e de dxdyplot

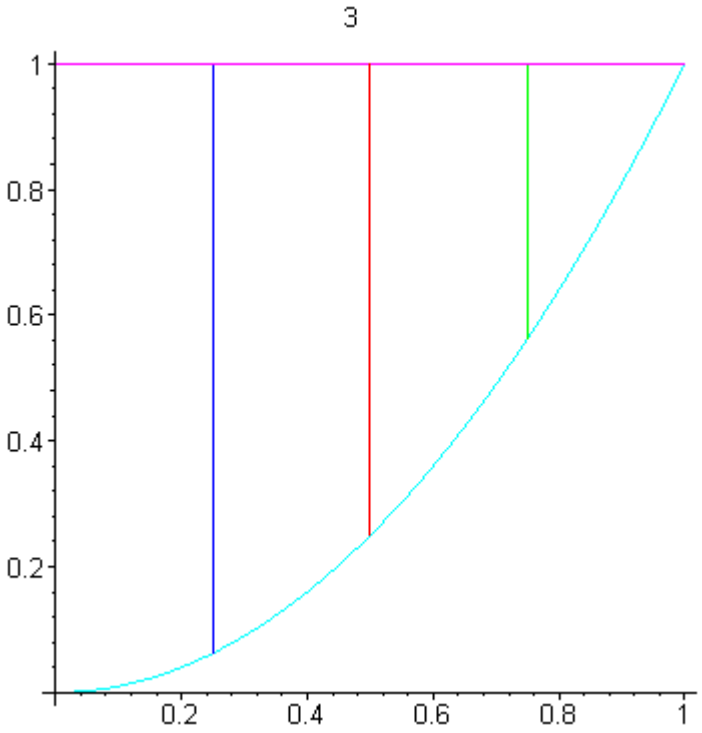
> **dydxplot(y=x^2..x+1,x=0..1,title=`1`);**



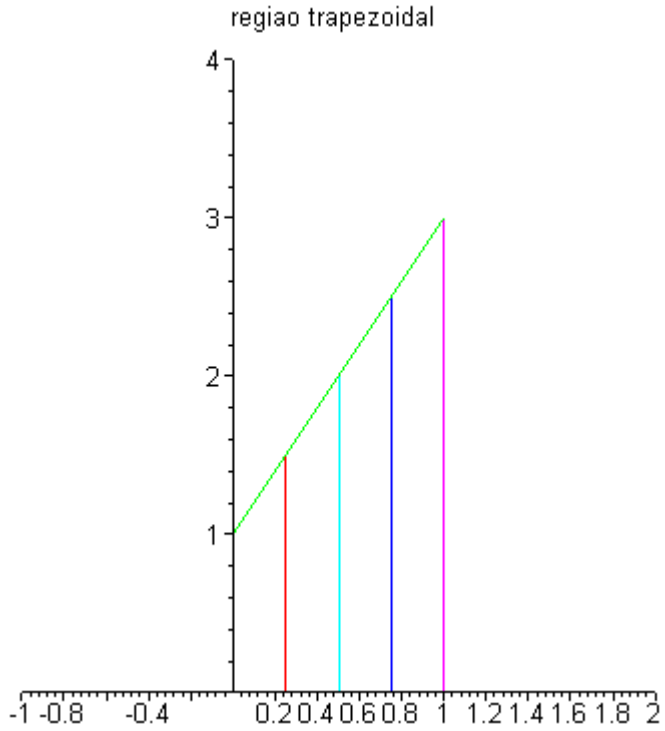
> **dxdyplot(x=0..sqrt(y),y=0..1, title=`2`);**



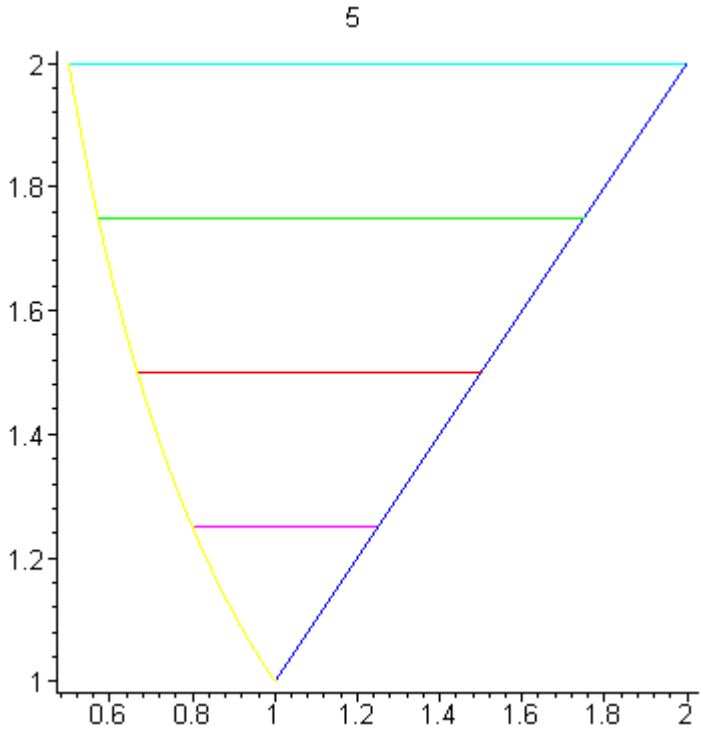
> **dydxplot(y=x^2..1,x=0..1, title=`3`);**



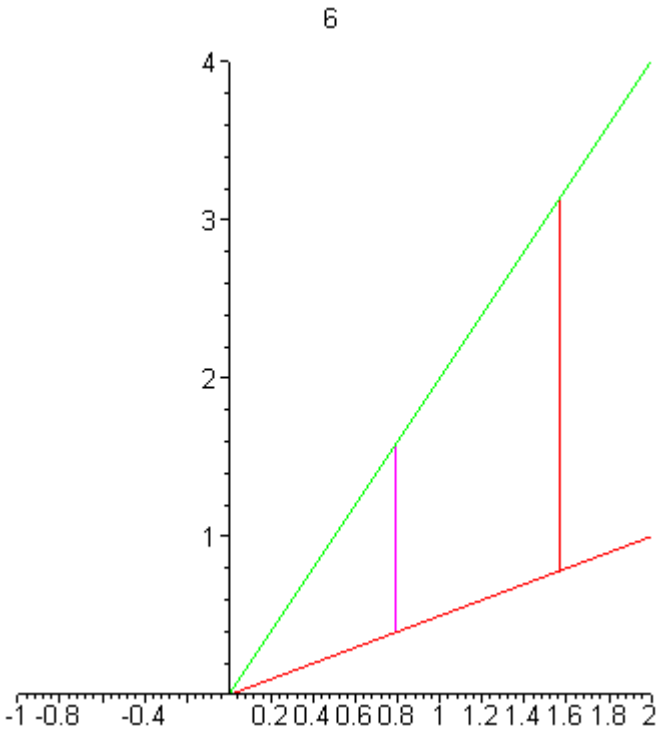
> **dydxplot(y=0..2*x+1,x=0..1,view=[-1..2,0..4], title=`regiao trapezoidal `);**



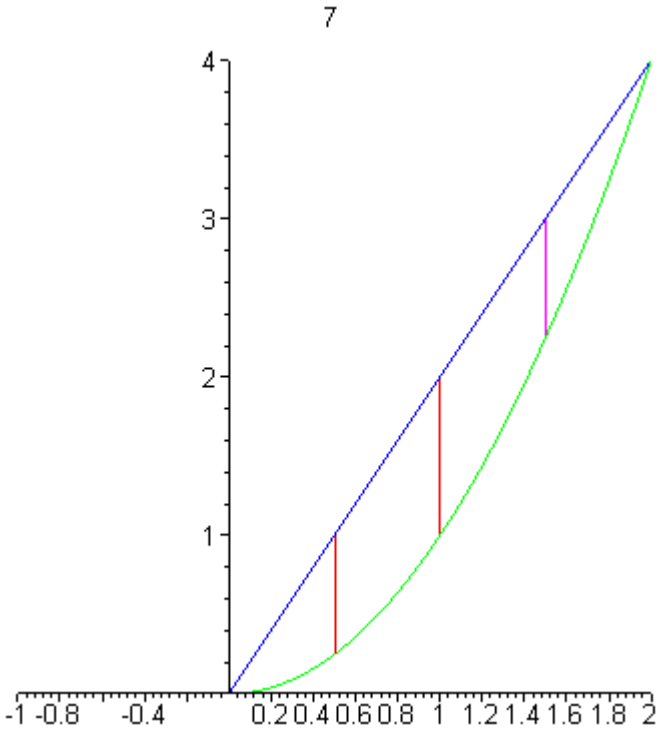
> **dxdyplot(x=y..1/y,y=1..2, title=`5`);**



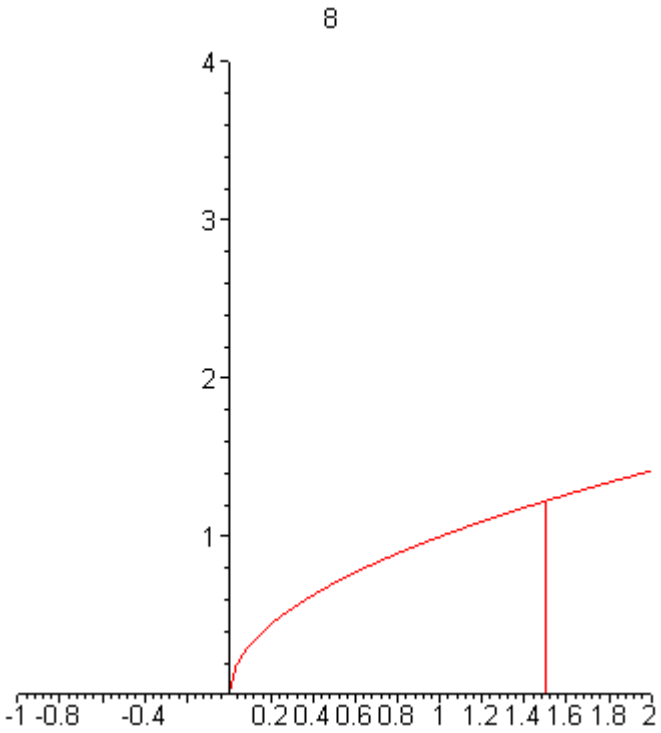
> **dydxplot(y=2*x..x/2,x=0..Pi,view=[-1..2,0..4], title=`6`);**



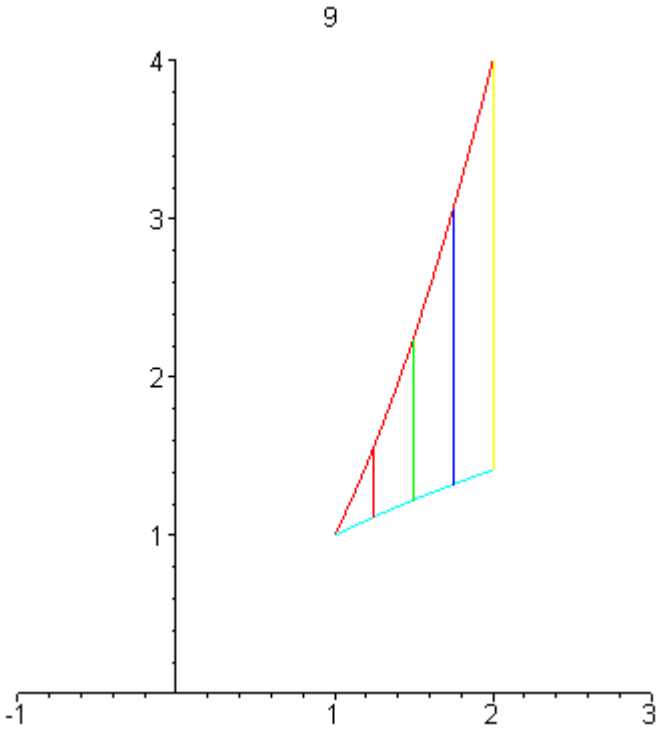
> **dydxplot(y=x^2..2*x,x=0..2,view=[-1..2,0..4], title=`7`);**



> **dydxplot(y=0..sqrt(x),x=0..6,view=[-1..2,0..4], title=`8`);**



> **dydxplot(y=sqrt(x)..x^2,x=1..2,view=[-1..3,0..4],title=`9`);**



>

O Procedimento (execute-o)

#Regiao do tipo I

```
> dydxplot:=proc()

> local a, b, c, d, g, h, i, j, f1, f2, opt_seq, p1, pset, v, ybound, xbound;

> if nargs < 2 then

> ERROR(`there must be at least two arguments`) fi;

> ybound:=args[1];

> xbound:=args[2];

> a:=op(1, rhs(xbound));

> b:=op(2, rhs(xbound));

> ### WARNING: semantics of type `string` have changed
if not type (xbound, string = range) then

> ERROR(`range expression for x is incorrect`) fi;

> ### WARNING: semantics of type `string` have changed
if not type (ybound, string = range) then

> ERROR(`range expression for y is not correct`) fi;

> if not (lhs(xbound) = 'x') then

> ERROR(`the second variable name must be x`) fi;

> if not (lhs(ybound) = 'y') then

> ERROR(`the first variable name must be y`) fi;

> if not type (evalf(a), numeric) or not type(evalf(b), numeric) then

> ERROR(`range limits for x are not real numbers` ) fi;

> c:=op(1, rhs(ybound));

> d:=op(2, rhs(ybound));

> g:=lhs(xbound);

> h:=lhs(ybound);

> f1:=unapply(c,g);
```

```
> f2:=unapply(d,g);

> p1:=plot({f1,f2},a..b);

> pset:={[v,f1(v),v=a..b],[v,f2(v),v=a..b]};

> for j from 1 to 5 do

> pset:=pset union

> {[a+(j-1)/4*(b-a),v*f2(a+(j-1)/4*(b-a))+(1-v)*f1(a+(j-1)/4*(b-a)),v=0..1]}; od;

> if nargs = 2 then

> p1:=plot(pset,axes=NORMAL,scaling=CONSTRAINED); fi;

> if nargs > 2 then

> opt_seq:=seq(args[i],i=3..nargs);

> p1:=plot(pset),opt_seq; fi;

> plots[display](p1);

> end:

# Regiao do tipo II

> dxdyplot:=proc()

> local a, b, c, d, g, h, i, j, f1, f2, opt_seq, p1, pset, v, ybound, xbound;

> if nargs < 2 then

> ERROR(`there must be at least two arguments`) fi;

> xbound:=args[1];

> ybound:=args[2];

> a:=op(1, rhs(ybound));

> b:=op(2, rhs(ybound));

> ### WARNING: semantics of type `string` have changed
if not type (xbound, string = range) then

> ERROR(`range expression for x is incorrect`) fi;

> ### WARNING: semantics of type `string` have changed
if not type (ybound, string = range) then
```

```
> ERROR(`range expression for y is not correct`) fi;
> if not (lhs(xbound) = 'x') then
> ERROR(`the second variable name must be x`) fi;
> if not (lhs(ybound) = 'y') then
> ERROR(`the first variable name must be y`) fi;
> if not type(evalf(a), numeric) or not type(evalf(b), numeric) then
> ERROR(`range limits for x are not real numbers` ) fi;
> c:=op(1, rhs(xbound));
> d:=op(2, rhs(xbound));
> g:=lhs(xbound);
> h:=lhs(ybound);
> f1:=unapply(c,h);
> f2:=unapply(d,h);
> pset:={[f1(h),h,h=a..b],[f2(h),h,h=a..b]};
> for j from 1 to 5 do
> pset:=pset union
> {[v*f2(a+(j-1)/4*(b-a))+(1-v)*f1(a+(j-1)/4*(b-a)),
> a+(j-1)/4*(b-a),v=0..1]}; od;
> if nargs = 2 then
> p1:=plot(pset,axes=NORMAL,scaling=CONSTRAINED); fi;
> if nargs >2 then
> opt_seq:=seq(args[i],i=3..nargs);
> p1:=plot(pset),opt_seq; fi;
> plots[display](p1);
> end:
```