



# Solides de révolution (Troisième partie de trois)

© Pierre Lantagne

Enseignant retraité du Collège de Maisonneuve

Ce document est une révision de celui produit en 2005. Au fil de cette série de documents Maple, le lecteur est introduit à un ensemble de procédures dédiées à la construction de solides de révolution. Ces procédures, écrites pour mes élèves, ont été réunies dans une bibliothèque que j'ai intitulée *riemann*. Ces procédures ont comme objectif principal de contribuer à l'amélioration chez l'élève leur capacité en perception spatiale. Dans ce troisième et dernier document de la série, on retrouvera de nombreux exemples documentés proposant une manière simple de construire des solides de révolution obtenus par la rotation d'une courbe pivotant autour d'un axe parallèle à l'axe des  $y$ .

Bonne lecture à tous !

\* Ce document Maple est exécutable avec la version 2020.2

## Initialisation

```
> restart;  
> with(plots,display,setoptions,setoptions3d,spacecurve);  
setoptions(size=[300,300],labels=[x,y],tickmarks=[12,12],axesfont=  
[TIMES,ROMAN,8],labelfont=[TIMES,ROMAN,8]);  
setoptions3d(size=[400,400],labels=[x,y,z],tickmarks=[4,4,4],  
axesfont=[TIMES,ROMAN,8],labelfont=[TIMES,ROMAN,8]);  
[display,setoptions,setoptions3d,spacecurve]
```

(1.1)

## Révolution autour d'un axe parallèle à l'axe des $y$

### Initialisation

La plupart des macro-commandes utilisées dans les exemples font parties de la bibliothèque *riemann* que j'ai écrits pour mes étudiants. Pour la séance Maple qui est en cours, vous devrez donc indiquer à Maple où aller lire les instructions composant les macro-commandes de cette bibliothèque. Vous devrez donc taper dans une zone de requêtes la requête suivante:

```
with(riemann);
```

Mais, préalablement, vous devrez indiquer à Maple le chemin à suivre pour accéder aux macro-commandes de cette bibliothèque, c'est-à-dire accéder au fichier contenant le code de ces macro-commandes. Il faut donc informer Maple du chemin menant au dossier contenant le fichier dans lequel il trouvera cette bibliothèque. Pour disposer des macro-commandes sur votre ordinateur, il vous faudra procéder à l'installation de deux fichiers « *riemann.ind* » et « *riemann.lib* » sur votre disque rigide: celui de votre ordinateur personnel et non pas celui du poste sur lequel vous travaillez au collège.

À partir de mon site internet, ....., cliquer sur l'onglet «NYB», localiser et télécharger les fichiers « *riemann.ind* » et « *riemann.lib* » et les déposer dans un sous-dossier de votre ordinateur. Je vous suggère de nommer ce sous-dossier « Bibliothèque Riemann » et un bon choix d'emplacement pour

ce dossier serait de le créer dans le dossier *Users* de votre installation Maple.

Ensuite, la façon d'informer Maple du chemin dont il est question consiste à **ajouter** à la variable d'environnement **libname** ce chemin. Mais d'abord, l'énoncé du chemin doit être une **chaîne de caractères** de la forme

```
"Nom_du_volume:\\Répertoire_1\\Répertoire_2\\...../Répertoire_contenant_le_fichier"
```

ATTENTION: La variable **libname** possède déjà une assignation qui a été faite au moment de l'installation du logiciel, soit l'assignation du chemin menant à la bibliothèque de base **maple.lib** contenu dans le dossier lib. Voyez:

```
> libname;
                                     "C:\Program Files\Maple 2020\lib"          (2.1.1)
```

Il ne faut surtout pas écraser le contenu de la variable **libname** par une nouvelle assignation, Maple ne saurait plus comment localiser les macro-commandes de la bibliothèque de base maple.lib. La façon dont il faut s'y prendre est la suivante:

```
libname := nouveau_chemin, libname;
```

Par exemple, l'assignation suivante est celle que je dois faire sur mon ordinateur personnel.

```
> libname := "C:\\Users\\plant\\Desktop\\Éléments site Maple au
cégep\\Downloads\\Documents Maple\\NYB/Bibliothèque riemann",
libname;
libname :=
    "C:\Users\plant\Desktop\Éléments site Maple au cégep\Downloads\Documents
    Maple\NYB/Bibliothèque riemann", "C:\Program Files\Maple 2020\lib"          (2.1.2)
```

On remarquera le dernier élément dans le résultat de cette assignation: c'est le chemin menant à la bibliothèque de base, c'est-à-dire que c'est le contenu qu'avait la variable *libname*. Cette façon de faire pour ajouter (concaténer) un nouveau terme à une séquence est valable quelque soit la variable en cause. Dans ce cas-ci, la variable en cause est la variable *libname*.

Maintenant, nous pouvons indiquer à Maple la manière d'accéder aux instructions des 18 macro-commandes de la bibliothèque *riemann*.

```
> with(riemann);
[coloriage, plot2ch, plot2cv, plot2dx, plot2dy, plot3dx, plot3dy, plotch, plotcoq, plotcoqanim,
plotcv, plotrecth, plotrectv, plotriemann, plotrotx, plotrotxanim, plotroty, plotrotyanim]          (2.1.3)
```

## Macro-commandes de y-révolution

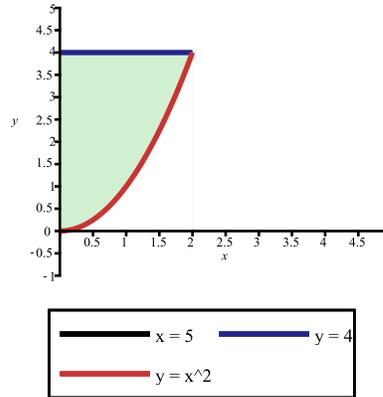
### plot2dy

La macro-commande

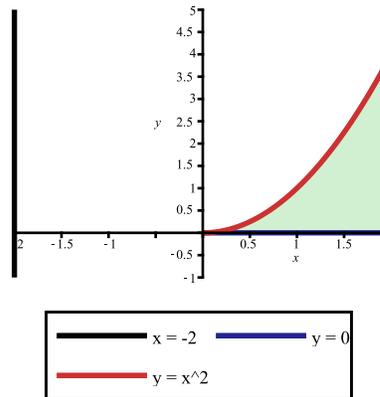
**plot2dy([f(x),g(x)], x=a..b, x=c, options)**

trace dans le plan cartésien la courbe d'équation  $y=f(x)$  en couleur **navy** et la courbe d'équation  $y=g(x)$  en couleur **orange** sur l'intervalle  $[a,b]$  de l'axe des  $x$ . La région du plan comprise entre ces deux courbes sur l'intervalle  $[a,b]$  est colorée en vert Maple. Les trois premiers arguments sont obligatoires. L'argument  $x=c$  permet de superposer l'axe de révolution.

```
> plot2dy([4,x^2],x=0..2,x=5);
```



```
> plot2dy([0,x^2],x=0..2,x=-2);
```



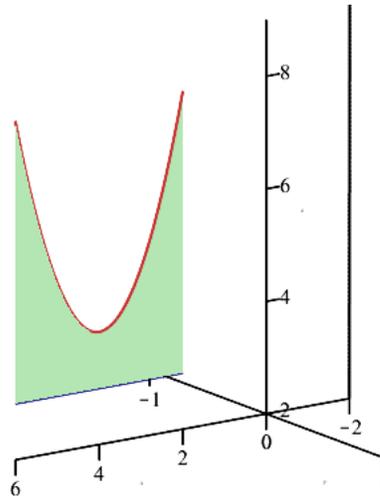
### plot3dy

La macro-commande

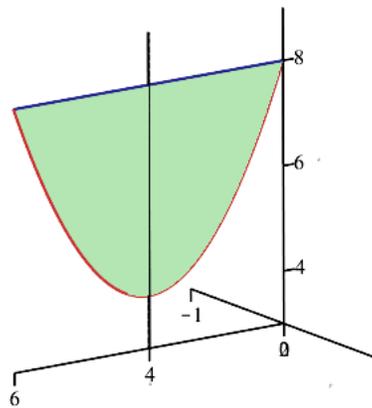
**plot3dy([f(x),g(x)], x=a..b , x=c, options)**

trace dans le plan  $xOz$  de l'espace la courbe d'équation  $z=f(x)$  en couleur navy et la courbe d'équation  $z=g(x)$  en couleur orange sur l'intervalle  $[a,b]$  de l'axe des  $x$ . La région de l'espace, dans le plan  $xOz$ , comprise entre ces deux courbes sur l'intervalle  $[a,b]$  est colorée en vert Maple. Le dernier argument obligatoire de cette procédure,  $z=c$ , permet de superposer dans le graphique l'axe de révolution.

```
> plot3dy([3,(x-4)^2+4],x=2..6,x=-2);
```



```
> plot3dy([8,(x-4)^2+4],x=2..6,x=4);
```



## plotroty

La macro-commande

**plotroty([f(x),g(x)], x=a..b, x=c, angle= $\theta$ , solide=mode, options)**

trace dans l'espace le solide engendré par la révolution d'un angle solide  $\theta$  autour de l'axe de révolution d'équation  $x = c$  de la surface de l'espace du plan  $xOz$  comprise entre la courbe d'équation  $y = f(x)$ , tracée en couleur navy, et la courbe d'équation  $y = g(x)$ , tracée en couleur orange, sur un intervalle  $[a,b]$  de l'axe des abscisses de l'espace.

L'angle  $\theta$  est en radians. Lorsque  $\theta > 0$ , le sens de la révolution est le sens trigonométrique. Un angle  $\theta < 0$  permet d'inverser le sens de la rotation. Si l'angle de rotation  $\theta$  est nul, seule la surface dans le plan  $xOz$  sera affichée.

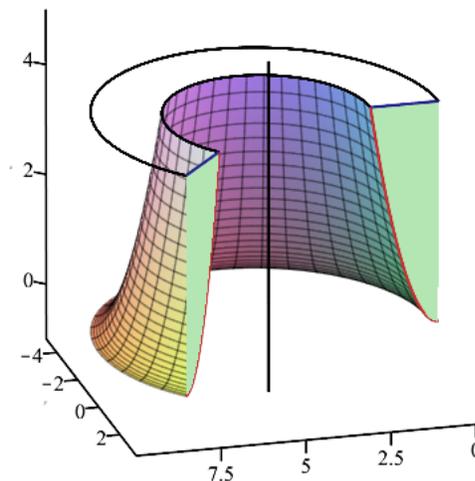
Les cinq premiers arguments sont obligatoires. Le cinquième argument « solide = mode » est obligatoire. En précisant la valeur "fermé" pour le mode, il s'ajoutera au tracé du solide les tracés des "panneaux" supérieurs et inférieurs parallèles au plan  $xOy$  de l'espace. La valeur "ouvert" permettra de

voir l'intérieur du solide de révolution. La version actuelle de la bibliothèque riemann ne prend pas en charge chaque partie d'une fonction définie par morceaux créée avec la macro-commande `piecewise`. En conséquence, l'argument « solide = ouvert » est inefficace pour les morceaux qui correspondront à des fonctions constantes: il y aura toujours le ou les tracés de ces panneaux supérieurs et inférieurs. Il est bien sûr possible d'éviter les tracés de tels panneaux. Il suffira de créer les tracés des solides de révolution en décomposant la fonction créée avec la macro-commande `piecewise` en des fonctions séparées. Ensuite, superposer ces tracés avec la macro-commande `display` de la bibliothèque `plots`.

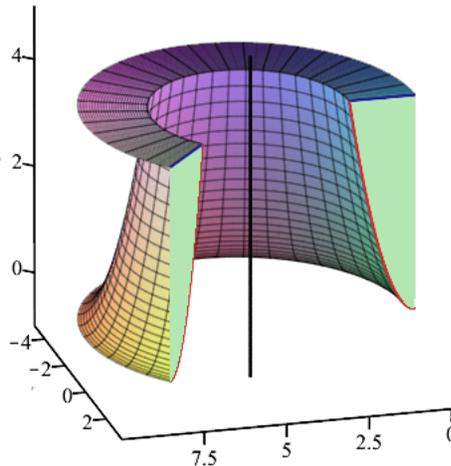
Lorsque la région possède des frontières parallèles au plan  $z = 0$  de l'espace, le tracé montre une trace pour suggérer la nature du solide engendré. Cette trace sera apparente avec l'argument « solide = ouvert ». En spécifiant l'argument « solide = fermé », cette trace se superposera au panneau supérieur et/ou inférieur et ne sera donc pas visible.

Remarque: Un `grid=[49,49]` est utilisé par défaut. Parfois, l'utilisation d'un grid plus fin sera nécessaire mais augmentera en conséquence le temps de calcul pour la création du graphique.

```
> plotroty([4,x^2],x=0..2,x=5,angle=5*Pi/4,solide=ouvert,
orientation=[75,70]);
```



```
> plotroty([4,x^2],x=0..2,x=5,angle=5*Pi/4,solide=fermé,
orientation=[75,70]);
```



## plotcoq

La macro-commande

**plotcoq(x=a, rayon=r, z=c..d, angle= $\theta$ , position=mode, options)**

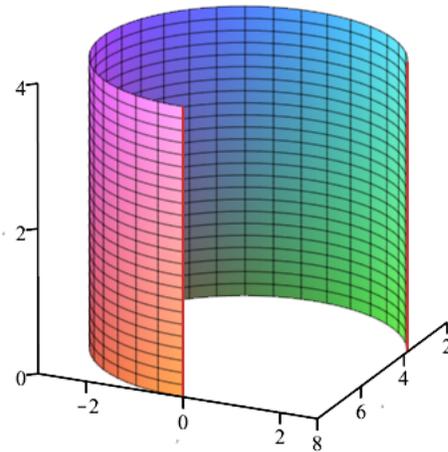
trace dans l'espace une coquille cylindrique engendré par la révolution d'un segment de droite parallèle à l'axe des cotes du plan  $xOz$ . L'axe de révolution est donné par l'équation  $x = a$ , l'angle solide de la révolution par  $\theta$  et la hauteur de la coquille cylindrique est déterminée par la longueur de l'intervalle  $[c, d]$ .

La valeur  $r$  que prendra le rayon de la coquille cylindrique sera déterminé par l'utilisateur en tenant compte de l'emplacement de l'axe de révolution et de la nature de la surface de révolution. Un tracé préliminaire de la surface dans l'espace pourra être utile dans ce cas. Lorsque  $\theta > 0$ , le sens de la construction de la coquille est le sens trigonométrique. Si  $\theta < 0$ , dans ce cas, le sens de la construction est le sens horaire.

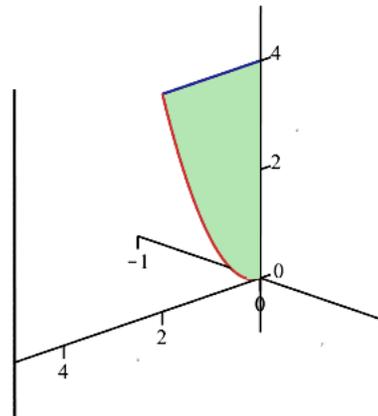
L'argument position doit prendre la modalité "droite" lorsque la partie principale de la surface de révolution est à la droite de l'axe et doit prendre la modalité "gauche" autrement. Considérer, comme d'habitude, la direction positive de l'axe des abscisses pour la "droite". Par exemple, 5 est à la droite de 3.

Remarque: Un `grid=[49,49]` est utilisé par défaut pour la création de la coquille.

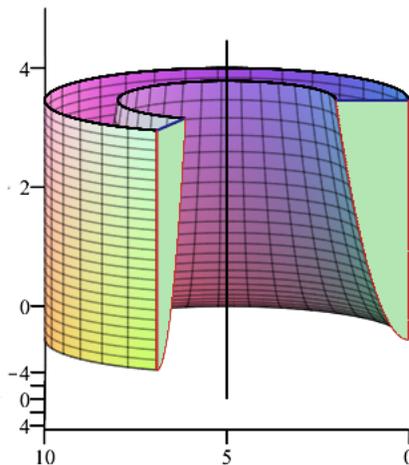
```
> plotcoq(x=5,rayon=3,z=0..4,angle=-11*Pi/8,position=droite,
orientation=[25,70]);
```



```
> plot3dy([4,x^2],x=0..2,x=5,orientation=[45,70]);
```

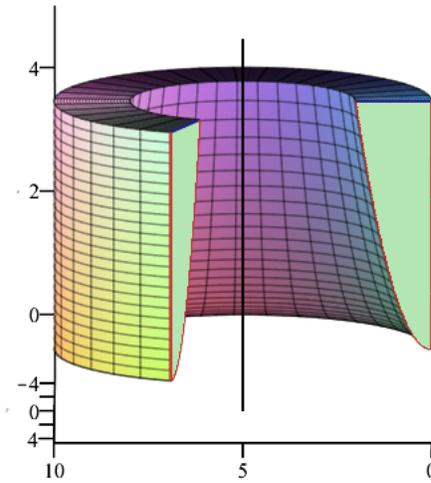


```
> A:=plotroty([4,x^2],x=0..2,x=5,angle=11*Pi/8, solide=ouvert):
> B:=plotcoq(x=5,rayon=5,z=0..4,angle=11*Pi/8,position=gauche):
> display(A,B,orientation=[90,80]);
```

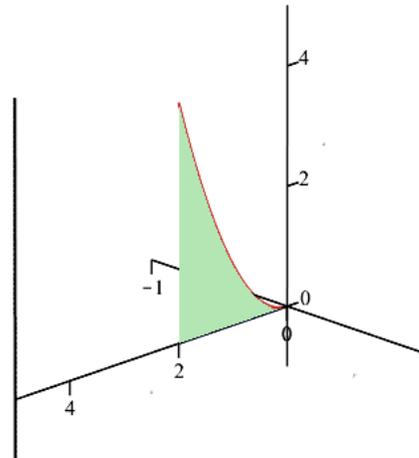


```
> A:=plotroty([4,x^2],x=0..2,x=5,angle=11*Pi/8, solide=fermé,
```

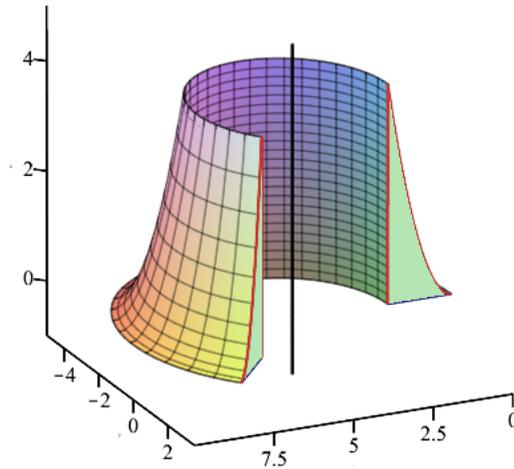
```
orientation=[75,70]):
display(A,B,orientation=[90,80]);
```



```
> plot3dy([0,x^2],x=0..2,x=5,orientation=[45,70]);
```



```
> A:=plotroty([0,x^2],x=0..2,x=5,angle=5*Pi/4,slide=fermé,grid=
[12,49]):
B:=plotcoq(x=5,rayon=3,z=0..4,angle=5*Pi/4,position=gauche):
display(A,B,orientation=[65,70]);
```



## plotrotyanim

La macro-commande

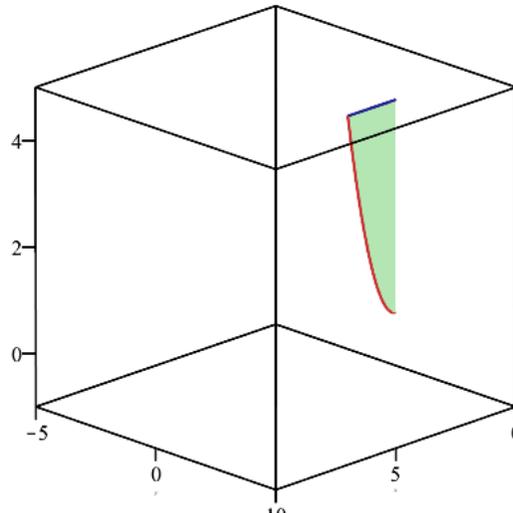
**plotrotyanim([f(x),g(x)], x=a..b, x=c, nimages=k, solide=mode, options)**

anime dans l'espace la création du solide de révolution engendré par la rotation d'un angle solide  $\theta$  autour de l'axe d'équation  $x = c$  de la surface comprise dans le plan  $xOz$  entre la courbe d'équation  $y = f(x)$  et la courbe d'équation  $y = g(x)$  sur l'intervalle  $[a,b]$  de l'axe des abscisses. La courbe  $y = f(x)$  sera tracée en couleur **navy** et la courbe d'équation  $y = g(x)$  le sera en couleur **orange**.

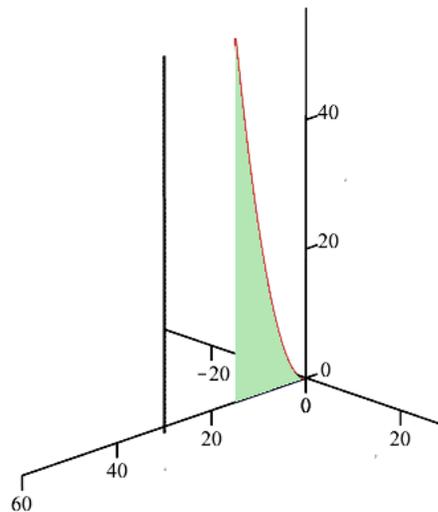
Tous les arguments de la macro-commande *plotrotyanim* sont ceux de la macro-commande *plotroty* sauf l'argument « angle =  $\theta$  » qui est remplacé par « nimages =  $k$  » où  $|k|$  précise le nombre d'images à bâtir pour l'animation. Lorsque  $k > 0$ , le sens de l'animation est le sens trigonométrique. Si  $k < 0$ , le sens de l'animation est le sens horaire.

Remarque: La création des séquences pour animer la création du solide de révolution requiert plusieurs secondes de calculs, parfois même de l'ordre de la minute même. Le réalisme rendu de la révolution de la surface en vaut tout de même la peine. Un grid=[49,49] est utilisé par défaut. L'utilisation d'un grid plus fin augmentera sensiblement le temps requis pour la création des surfaces de chaque image. Donc, le temps nécessaire à la création du solide est en fonction du nombre d'images et/ou du grid précisé par l'utilisateur.

```
> plotrotyanim([4,x^2],x=0..2,x=5,nimages=24,solide=ouvert,axes=boxed,orientation=[45,70]);
```



```
> plotrotyanim([0,1/4*x^2],x=0..15,x=30,nimages=24,solide=fermé,
orientation=[45,70],axes=normal);
```



## plotcoqanim

La macro-commande

**plotcoqanim(x=a, rayon=r, z=c..d, nimages=k, position=mode, options)**

anime dans l'espace la création d'une coquille cylindrique engendré par la révolution d'un segment de droite (du plan  $xOz$ ) parallèle à l'axe des cotes du plan  $xOz$ . L'axe de rotation est donné par l'équation  $x = a$ , l'angle solide de la rotation par  $\theta$  et la hauteur de la coquille cylindrique est déterminée par la longueur de l'intervalle  $[c,d]$ .

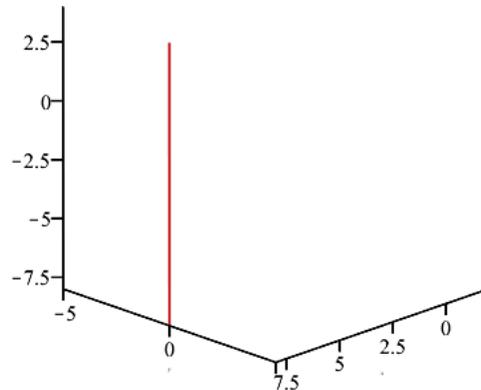
Tous les arguments de la macro-commande *plotcoqanim* sont ceux de la macro-commande *plotcoq* sauf l'argument « angle =  $\theta$  » qui est remplacé par « nimages =  $k$  » où  $|k|$  précise le nombre d'images à bâtir pour l'animation. Lorsque  $k > 0$ , le sens de l'animation est le sens trigonométrique. Si  $k < 0$ , le sens de l'animation est le sens horaire.

Rappelons que l'argument position doit prendre la modalité "droite" lorsque la partie principale de la surface de révolution est à la droite de l'axe et doit prendre la modalité "gauche" autrement.

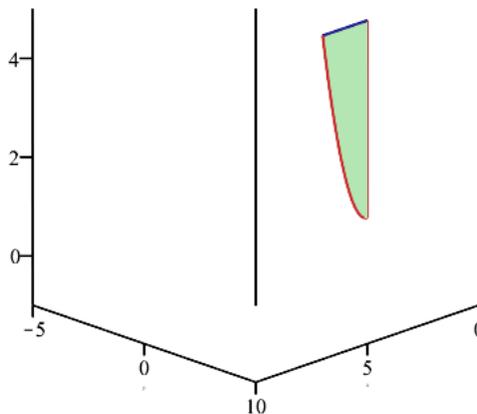
Considérer, comme d'habitude, la direction positive de l'axe des abscisses pour la "droite". Par exemple, 5 est à la droite de 3.

Remarque: Un `grid=[49,49]` est utilisé par défaut pour la création de chaque séquence de l'animation de la coquille.

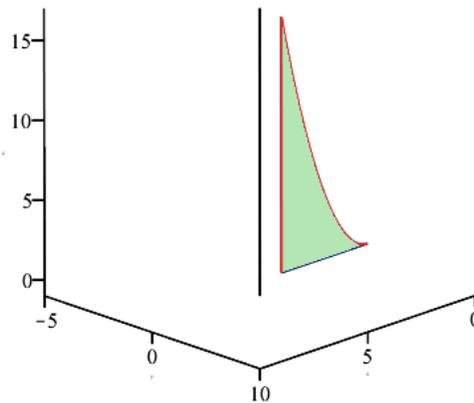
```
> plotcoqanim(x=3,rayon=5,z=-8..4,nimages=24,position=droite,  
orientation=[45,70]);
```



```
> A:=plotrotyanim([4,x^2],x=0..2,x=5,nimages=24,solide=fermé):  
B:=plotcoqanim(x=5,rayon=5,z=0..4,nimages=24,position=gauche):  
> display(A,B,orientation=[45,70]);
```



```
> A:=plotrotyanim([0,x^2],x=0..4,x=5,nimages=24,solide=fermé):  
B:=plotcoqanim(x=5,rayon=1,z=0..16,nimages=24,position=gauche):  
> display(A,B,orientation=[45,70]);
```



## Exemples

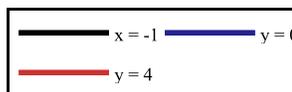
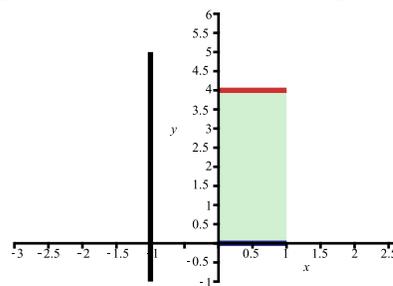
Chaque solide sera développé en

- représentant d'abord la région dans le plan cartésien
- représentant ensuite la région dans l'espace
- créant le solide avec un angle donné  $\theta$  de révolution
- animant finalement la création de ce solide.

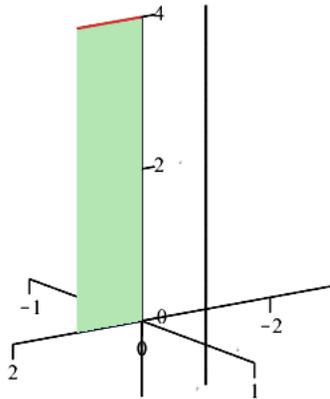
### Exemple 1

Soit le solide de révolution obtenu par la rotation de la surface comprise entre l'axe des  $x$  et la courbe d'équation  $y=f(x)=4$  limitée par l'intervalle  $[0,1]$ . L'axe de rotation est la droite d'équation  $x=-1$ .

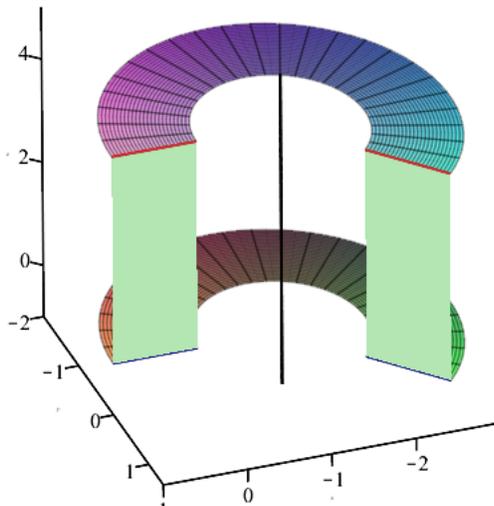
```
> f:=x->0:
g:=x->4:
plot2dy([f(x),g(x)],x=0..1,x=-1,view=[-3..2.6,-1..6]);
```



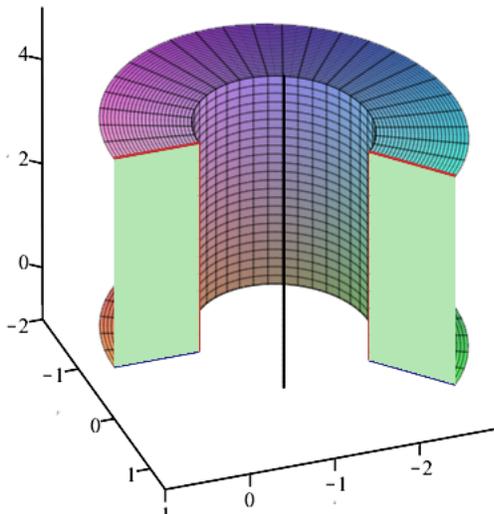
```
> plot3dy([f(x),g(x)],x=0..1,x=-1,view=[-3..2,-1..1,-1..4],axes=
normal);
```



```
> A:=plotroty([f(x),g(x)],x=0..1,x=-1,angle=-5*Pi/4,solide=fermé):
display(A,orientation=[70,60],axes=framed);
```



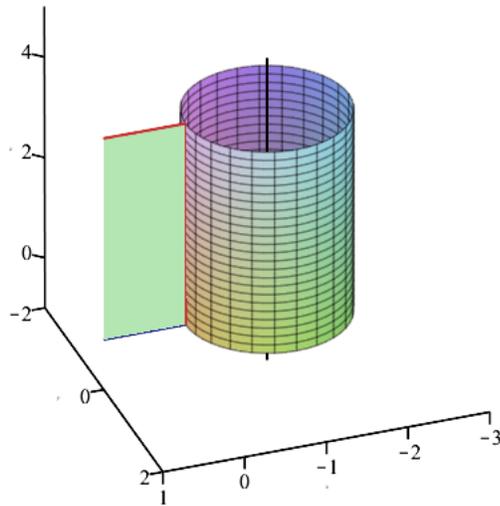
```
> B:=plotcoq(x=-1,rayon=1,z=0..4,angle=-5*Pi/4,position=droite):
display(A,B,orientation=[70,60],axes=framed);
```



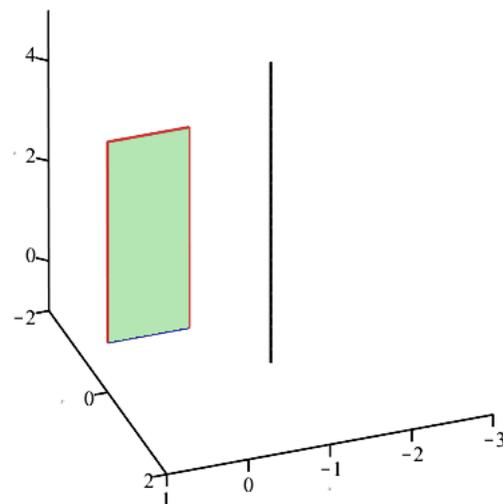
```
> A:=plotrotyanim([f(x),g(x)],x=0..1,x=-1,nimages=-24,solide=
```

fermé):

```
B:=plotcoq(x=-1,rayon=1,z=0..4,angle=2*Pi,position=droite):  
display(A,B,orientation=[70,60]);
```



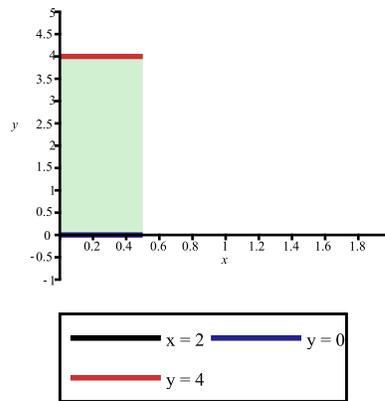
```
> B:=plotcoqanim(x=-1,rayon=1,z=0..4,nimages=-24,position=droite):  
C:=plotcoqanim(x=-1,rayon=2,z=0..4,nimages=-24,position=droite):  
display(A,B,C,orientation=[70,60]);
```



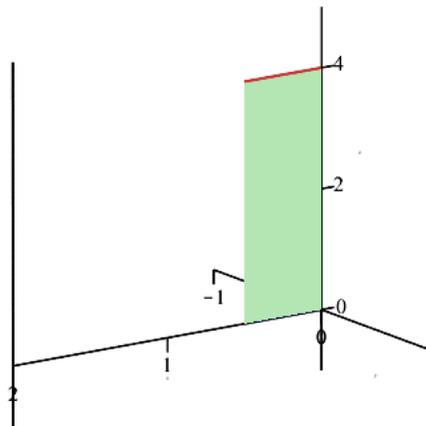
### Exemple 2

Soit le solide de révolution obtenu par la rotation de la surface comprise entre l'axe des  $x$  et la courbe d'équation  $y=f(x)=4$  limitée par l'intervalle  $\left[0, \frac{1}{2}\right]$ . L'axe de rotation est la droite d'équation  $x=2$ .

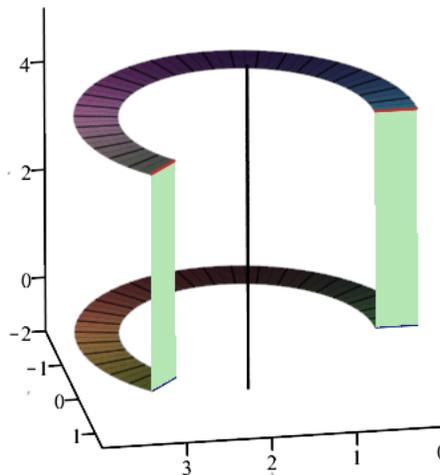
```
> f:=x->0:  
g:=x->4:  
plot2dy([f(x),g(x)],x=0..1/2,x=2);
```



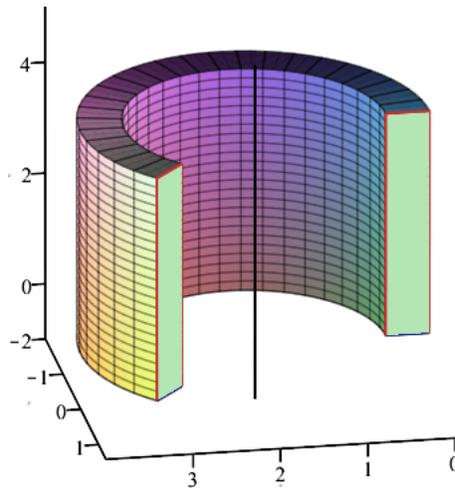
```
> plot3dy([f(x),g(x)],x=0..1/2,x=2,axes=normal);
```



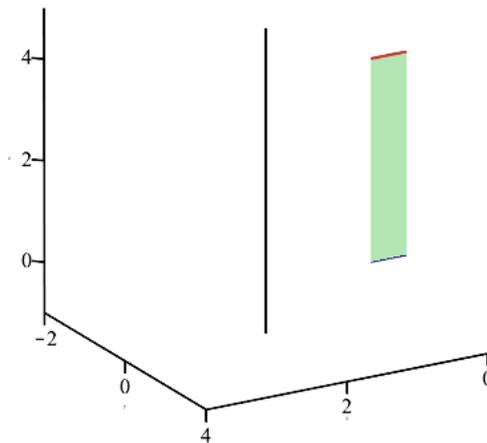
```
> A:=plotroty([f(x),g(x)],x=0..1/2,x=2,angle=5*Pi/4,solide=fermé):
display(A,orientation=[80,70]);
```



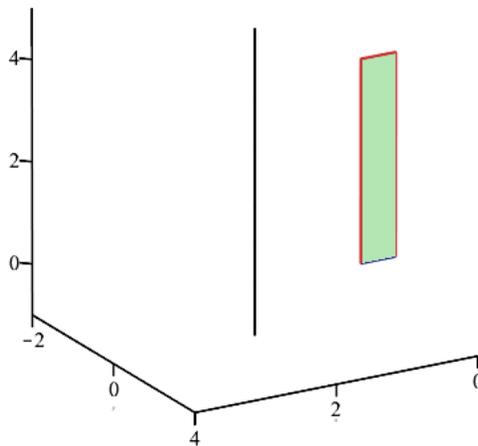
```
> B:=plotcoq(x=2,rayon=2,z=0..4,angle=5*Pi/4,position=gauche):
C:=plotcoq(x=2,rayon=3/2,z=0..4,angle=5*Pi/4,position=gauche):
display(A,B,C,orientation=[80,70]);
```



```
> A:=plotrotyanim([f(x),g(x)],x=0..1/2,x=2,nimages=24,solide=
fermé):
display(A,insequence=true,orientation=[60,70]);
```



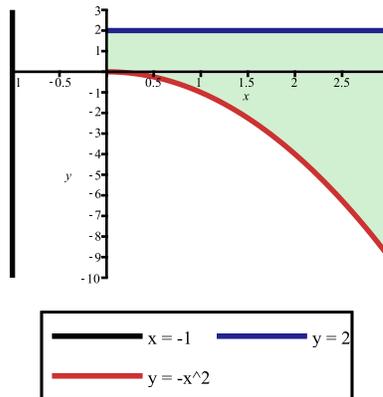
```
> B:=plotcoqanim(x=2,rayon=2,z=0..4,nimages=24,position=gauche):
C:=plotcoqanim(x=2,rayon=3/2,z=0..4,nimages=24,position=gauche):
> display(A,B,C,orientation=[60,70]);
```



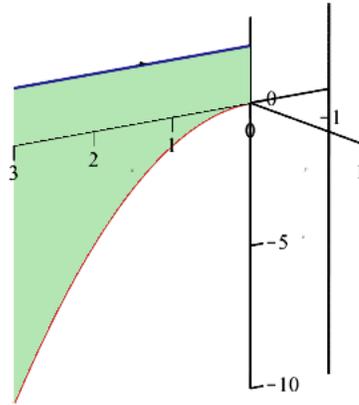
### Exemple 3

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = 2$  et la courbe d'équation  $y = f(x) = -x^2$  limitée par l'intervalle  $[0, 3]$ . L'axe de rotation est la droite d'équation  $x = -1$ .

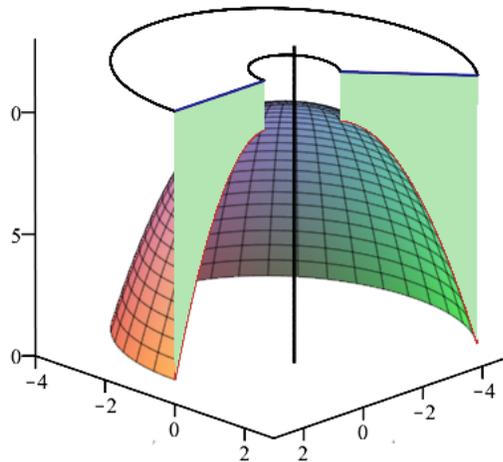
```
> f:=x->2:
g:=x->-x^2:
plot2dy([f(x),g(x)],x=0..3,x=-1);
```



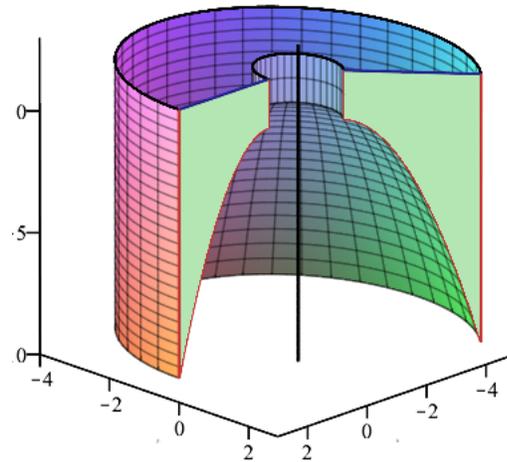
```
> plot3dy([f(x),g(x)],x=0..3,x=-1,axes=normal);
```



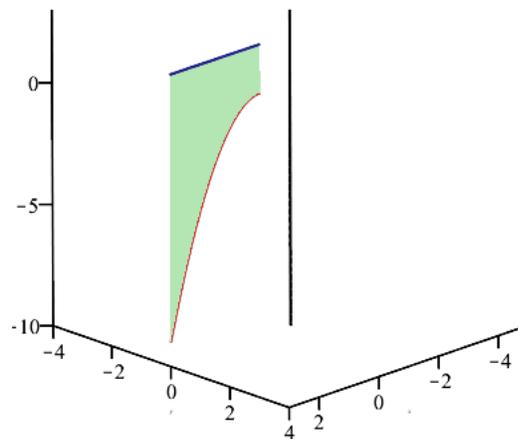
```
> A:=plotroty([f(x),g(x)],x=0..3,x=-1,angle=-5*Pi/4,solide=ouvert)
:
display(A,orientation=[45,70]);
```



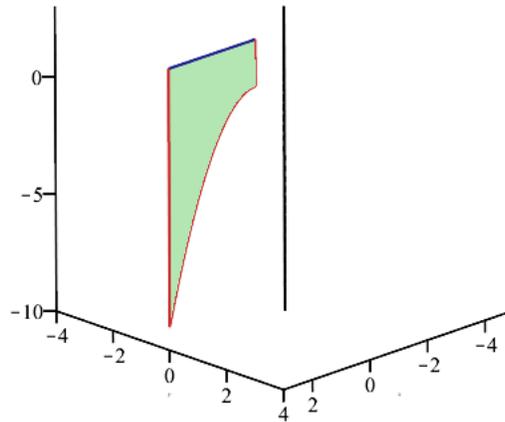
```
> B:=plotcoq(x=-1,rayon=4,z=f(3)..g(3),angle=-5*Pi/4,position=
droite):
C:=plotcoq(x=-1,rayon=1,z=f(0)..g(0),angle=-5*Pi/4,position=
droite,grid=[3,25]):
display(A,B,C,orientation=[45,70]);
```



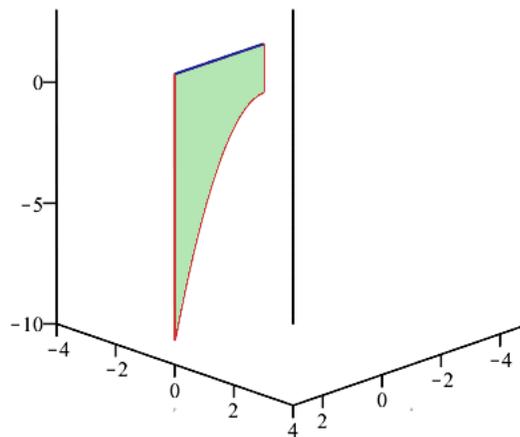
```
> A:=plotrotyanim([f(x),g(x)],x=0..3,x=-1,nimages=-24,solide=
ouvert):
display(A,insequence=true,orientation=[45,70]);
```



```
> B:=plotcoqanim(x=-1,rayon=4,z=f(3)..g(3),nimages=-24,position=
droite):
C:=plotcoqanim(x=-1,rayon=1,z=f(0)..g(0),nimages=-24,position=
droite,grid=[3,25]):
display(A,B,C,orientation=[45,70]);
```



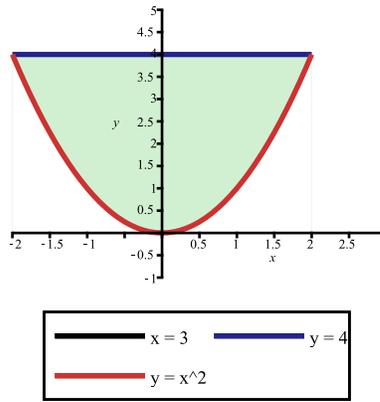
```
> A:=plotrotyanim([f(x),g(x)],x=0..3,x=-1,nimages=-24,solide=
fermé):
display(A,B,C,orientation=[45,70]);
```



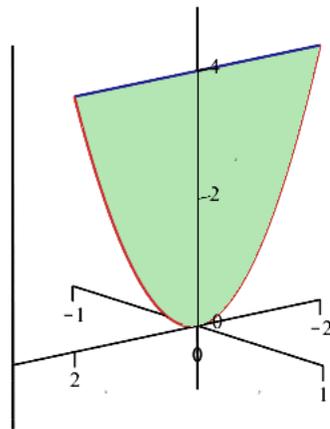
#### Exemple 4

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = 4$  et la courbe d'équation  $y = x^2$  limitée par l'intervalle  $[-2, 2]$ . L'axe de rotation est la droite d'équation  $x = 3$ .

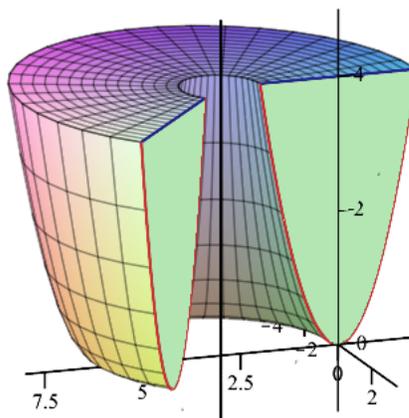
```
> f:=x->4:
g:=x->x^2:
> plot2dy([f(x),g(x)],x=-2..2,x=3);
```



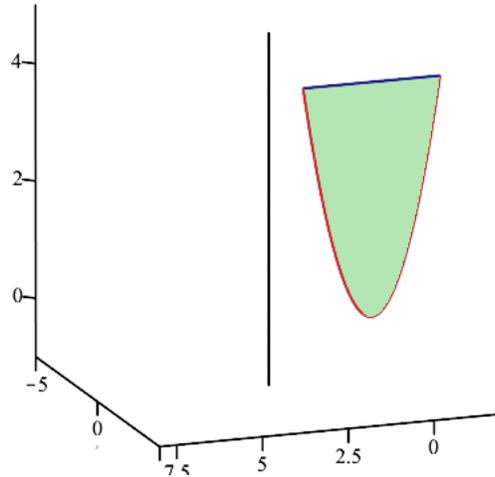
```
> plot3dy([f(x),g(x)],x=-2..2,x=3,axes=normal,orientation=[50,75])
;
```



```
> plotroty([f(x),g(x)],x=-2..2,x=3,angle=5*Pi/4,solide=fermé,grid=
[15,25],orientation=[70,75],axes=normal);
```



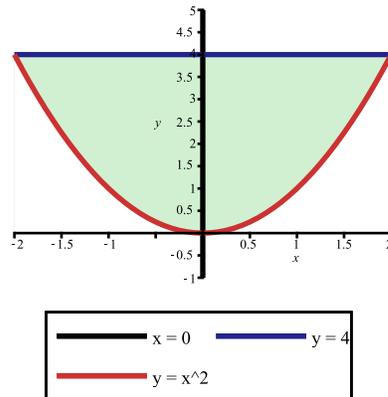
```
> plotrotyanim([f(x),g(x)],x=-2..2,x=3,nimages=24,solide=fermé,
orientation=[70,75]);
```



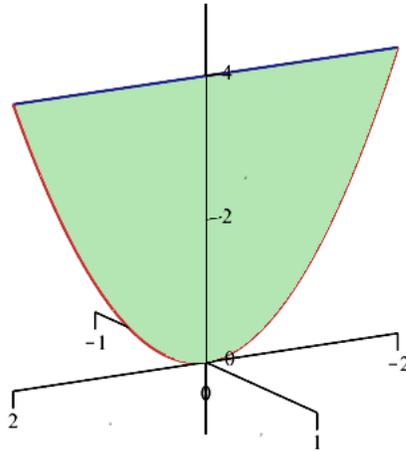
### Exemple 5

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = 4$  et la courbe d'équation  $y = x^2$  limitée par l'intervalle  $[-2, 2]$ . L'axe de rotation est la droite d'équation  $x = 3$ .

```
> f:=x->4:
g:=x->x^2:
plot2dy([f(x),g(x)],x=-2..2,x=0);
```



```
> plot3dy([f(x),g(x)],x=-2..2,x=0,orientation=[60,75],axes=normal)
;
```



Il suffit d'un angle  $\theta = \pi$  pour générer complètement le solide. En effet, les régions limitées par l'intervalle  $[-2, 0]$  et l'intervalle  $[0, 2]$  contribuent respectivement à la création de la moitié de ce solide de révolution.

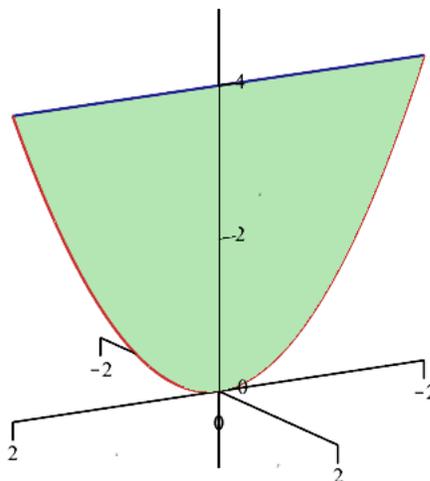
L'animation suivante va vous permettre de voir justement la contribution de chacune de ces deux sous-régions à la création de ce solide lorsque que l'angle de rotation est  $\theta = 2\pi$ .

```
> plotrotyanim([f(x),g(x)],x=-2..2,x=0,nimages=24,solide=ouvert,
orientation=[60,75],axes=normal);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

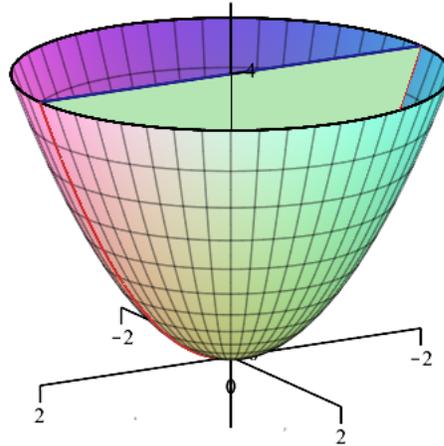


Si on conservait la totalité de la région limitée par l'intervalle  $[-2, 2]$ , une rotation d'angle  $\theta = \pi$  permettrait la création de ce solide de révolution.

```
> plotroty([f(x),g(x)],x=-2..2,x=0,angle=Pi,solide=ouvert,axes=
normal,orientation=[60,75]);
```

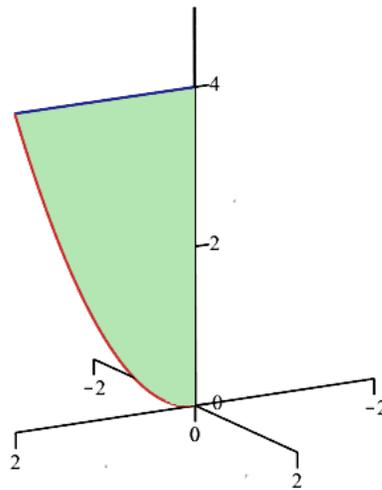
*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

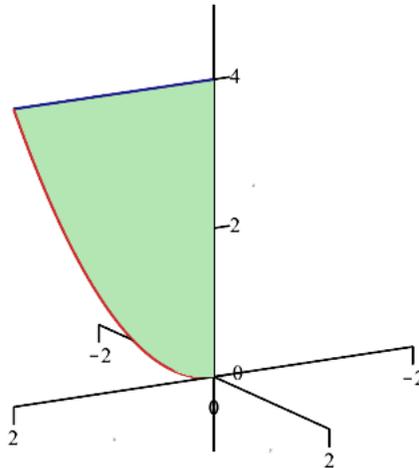


Puisque l'axe de rotation chevauche la région et aussi puisque l'axe de rotation est une axe de symétrie, il suffit de considérer seulement la région limitée par l'intervalle  $[0, 2]$  comme région génératrice de ce solide.

```
> plot3dy([f(x),g(x)],x=0..2,x=0,orientation=[60,75],axes=normal,  
view=[-2..2,-2..2,0..5]);
```

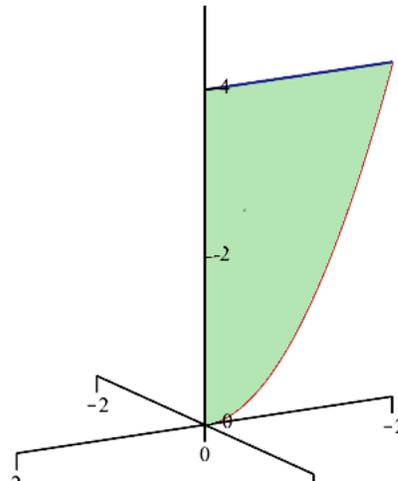


```
> plotrotyanim([f(x),g(x)],x=0..2,x=0,nimages=-24,solide=fermé,  
orientation=[60,75],axes=normal);
```

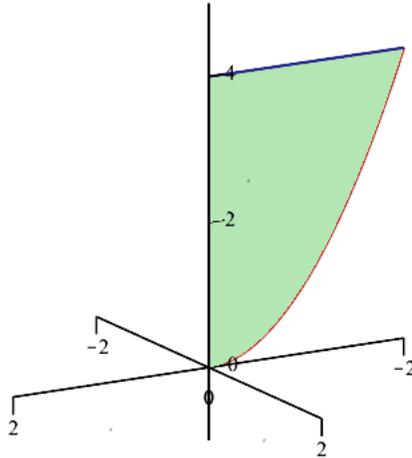


La région limitée par l'intervalle  $[-2, 0]$  pourrait tout aussi bien faire l'affaire. Nous pouvons donc aussi bien considérer seulement la région limitée par l'intervalle  $[-2, 0]$  comme région génératrice.

```
> plot3dy([f(x),g(x)],x=-2..0,x=0,orientation=[60,75],axes=normal,
view=[-2..2,-2..2,0..5]);
```



```
> plotrotyanim([f(x),g(x)],x=-2..0,x=0,nimages=24,solide=fermé,
orientation=[60,75],axes=normal);
```



### Exemple 6

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = 0$  et la courbe d'équation

$$y = g(x) = \begin{cases} x^2 & -x \leq 0 \text{ and } x < 2 \\ 4 & 2 - x \leq 0 \text{ and } x - 4 \leq 0 \end{cases}$$

limitée par l'intervalle  $[0, 4]$ . L'axe de rotation est la droite d'équation  $x = 1$ .

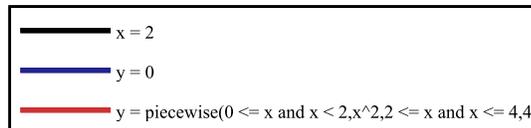
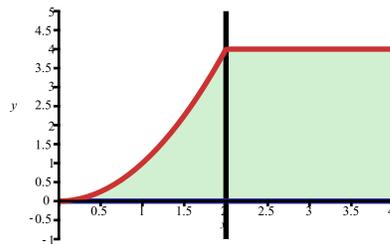
```
> f:=x->0;
g:=x->piecewise(x>=0 and x<2,x^2,x>=2 and x<=4,4);
'g'(x)=g(x);
```

$$f := x \mapsto 0$$

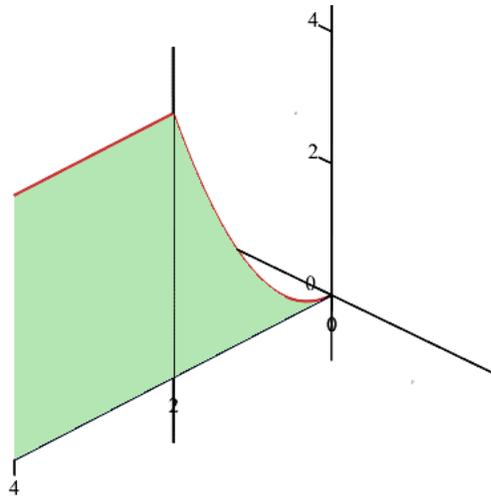
$$g(x) = \begin{cases} x^2 & 0 \leq x < 2 \\ 4 & 2 \leq x \leq 4 \end{cases}$$

(2.3.6.1)

```
> plot2dy([f(x),g(x)],x=0..4,x=2);
```



```
> plot3dy([f(x),g(x)],x=0..4,x=2,orientation=[44,60]);
```

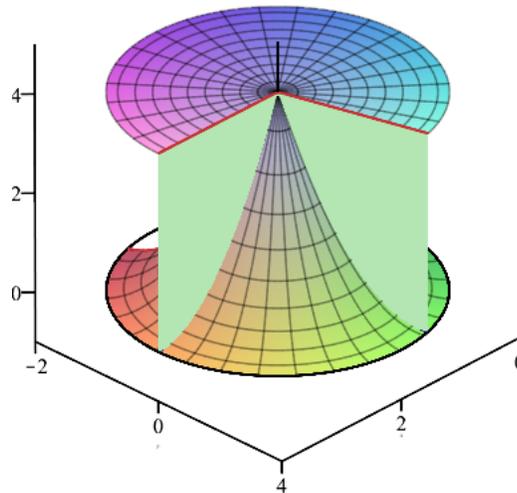


```
> plotroty([f(x),g(x)],x=0..4,x=2,angle=-17*Pi/12,solide=ouvert,
grid=[60,60],orientation=[44,60]);
```

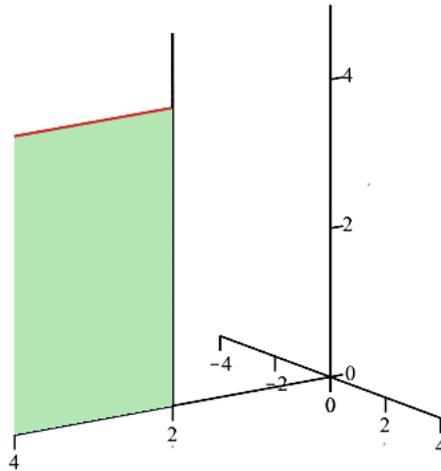
*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

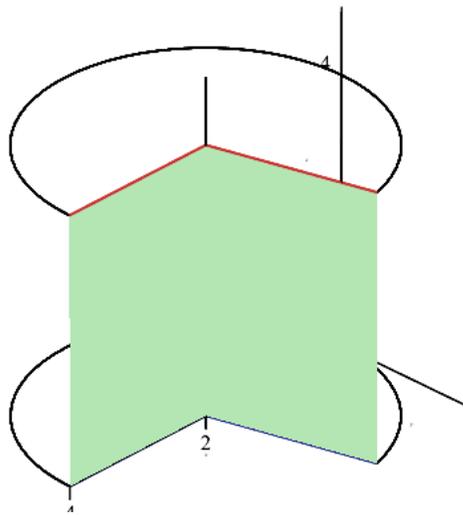
*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*



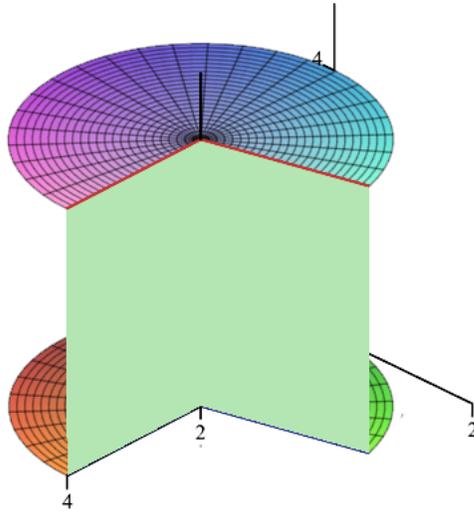
```
> plot3dy([f(x),g(x)],x=2..4,x=2,orientation=[44,60],numpoints=
120,view=[0..4,-4..4,0..5]);
```



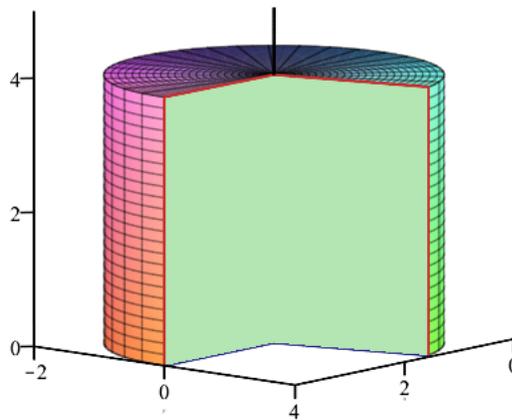
```
> A:=plotroty([f(x),g(x)],x=2..4,x=2,angle=-17*Pi/12,solide=
ouvert):
display(A,axes=normal,orientation=[44,60],view=[0..4,-2..2,0..5]
);
```



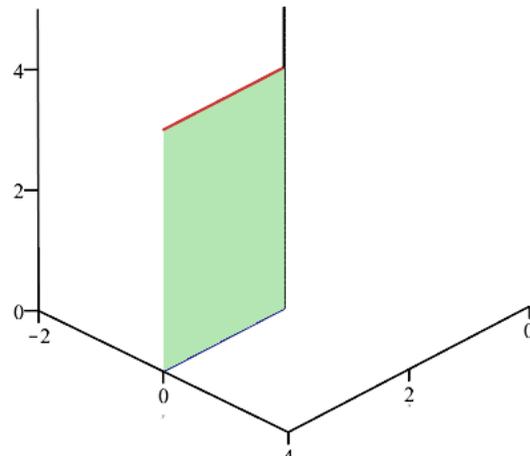
```
> A:=plotroty([f(x),g(x)],x=2..4,x=2,angle=-17*Pi/12,solide=fermé)
:
display(A,axes=normal,grid=[60,60],orientation=[44,60],view=[0.
.4,-2..2,0..5]);
```



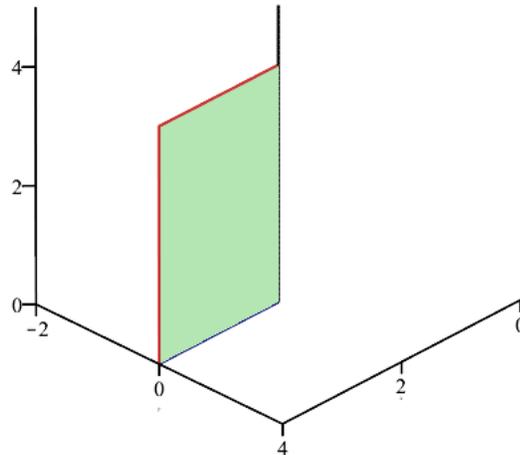
```
> B:=plotcoq(x=2,rayon=2,z=0..4,angle=-17*Pi/12,position=droite):
display(A,B,orientation=[44,60],orientation=[40,80],view=[0..4,
-2..2,0..5]);
```



```
> A:=plotrotyanim([f(x),g(x)],x=2..4,x=2,nimages=-24,solide=fermé)
:
display(A,grid=[60,60],insequence=true,orientation=[44,60],view=
[0..4,-2..2,0..5]);
```



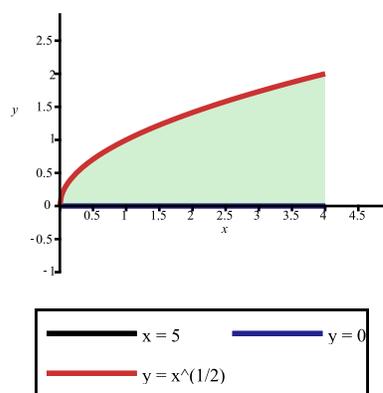
```
> B:=plotcoqanim(x=2,rayon=2,z=0..4,nimages=-24,position=droite):
display(A,B,orientation=[44,60],view=[0..4,-2..2,0..5]);
```



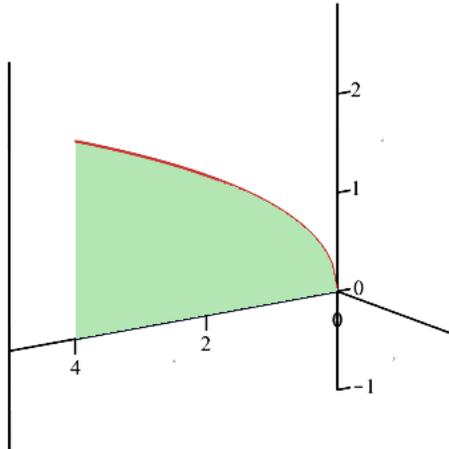
### Exemple 7

Soit le solide de révolution obtenu par la rotation de la surface comprise entre l'axe des  $x$  et la courbe d'équation  $y=f(x)=\sqrt{x}$  limitée par l'intervalle  $[0, 4]$ . L'axe de rotation est la droite d'équation  $x=5$ .

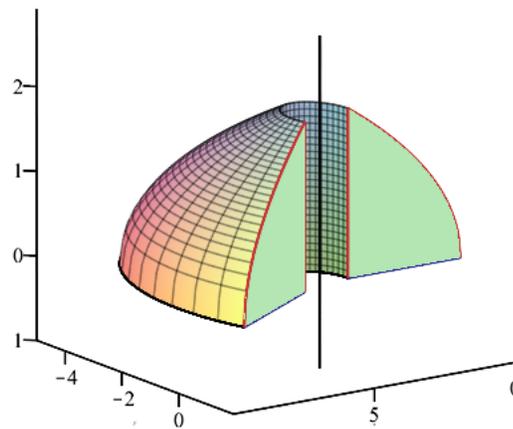
```
> f:=x->0:
g:=x->sqrt(x):
plot2dy([f(x),g(x)],x=0..4,x=5,scaling=constrained);
```



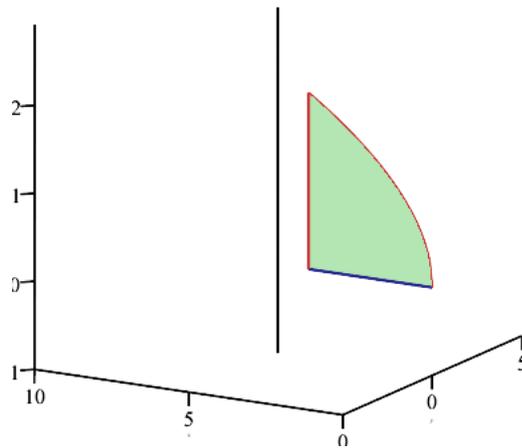
```
> plot3dy([f(x),g(x)],x=0..4,x=5,axes=normal);
```



```
> A:=plotroty([f(x),g(x)],x=0..4,x=5,angle=9*Pi/8,solide=ouvert):  
B:=plotcoq(x=5,rayon=1,z=0..2,angle=9*Pi/8,position=gauche):  
display(A,B,orientation=[55,75]);
```



```
> A:=plotrotyanim([f(x),g(x)],x=0..4,x=5,nimages=24,solide=ouvert)  
:  
B:=plotcoqanim(x=5,rayon=1,z=0..2,nimages=24,position=gauche):  
display(A,B,orientation=[-120,75,-180]);
```

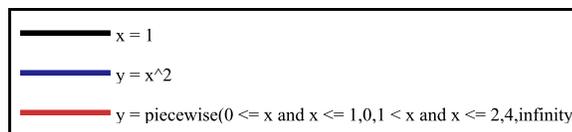
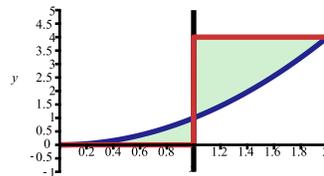


### Exemple 8

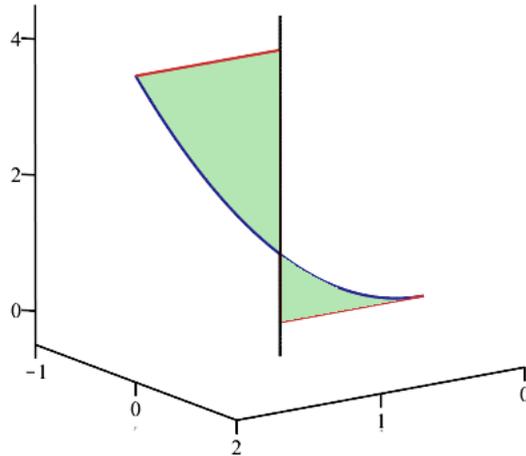
Définir la région à l'aide de la macro-commande *piecewise* donne un résultat plus ou moins bon pour la visualisation du solide dans l'espace et pour l'animation de ce solide également. C'est le cas lorsque la fonction créée avec la macro-commande *piecewise* est une fonction qui n'est pas continue. Aux points de discontinuité, le polygone (la surface verte) épousera approximativement la courbe (avec un bon zoom avant). Dans de tels cas, il sera préférable de procéder par la superposition de solides sans utiliser la macro-commande *piecewise*.

Afin que vous puissiez voir la différence, traitons d'abord la création du solide suivant avec la macro-commande *piecewise*.

```
> f:=x->x^2:
g:=x->piecewise(x>=0 and x<=1,0,x>1 and x<=2,4,infinity):
plot2dy([f(x),g(x)],x=0..2,x=1,size=[250,250]);
```



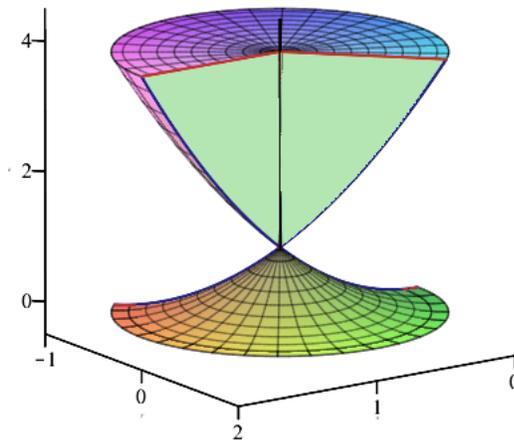
```
> plot3dy([f(x),g(x)],x=0..2,x=1,grid=[80,80],numpoints=400,axes=
framed,orientation=[55,75],view=[0..2,-1..1,-0.5..4.5]);
```



```
> plotroty([f(x),g(x)],x=0..2,x=1,angle=-5*Pi/4,solide=ouvert,
grid=[60,60],numpoints=400,orientation=[55,75],view=[0..2,-1..1,
-0.5..4.5]);
```

*Remarque:*

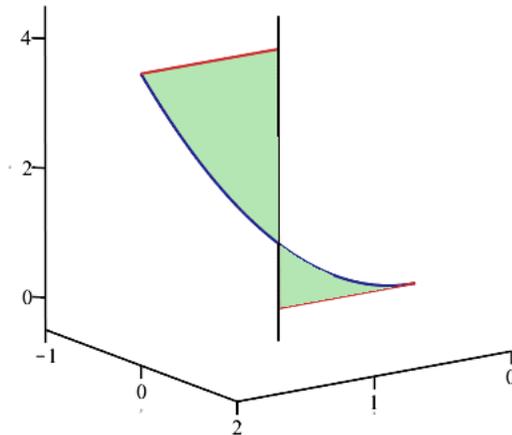
*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*



```
> plotrotyanim([f(x),g(x)],x=0..2,x=1,nimages=-24,solide=ouvert,
grid=[60,60],numpoints=400,orientation=[55,75],view=[0..2,-1..1,
-0.5..4.5]);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*



Maintenant, pour un meilleur rendu du solide ainsi que son animation, décomposons donc la fonction définies par morceaux.

```
> f1:=x->x^2;
g1:=x->0;
f2:=x->x^2;
g2:=x->4;
```

$$f1 := x \mapsto x^2$$

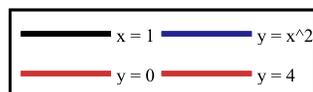
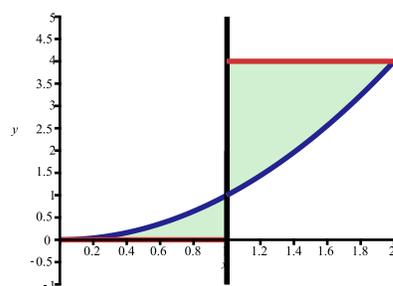
$$g1 := x \mapsto 0$$

$$f2 := x \mapsto x^2$$

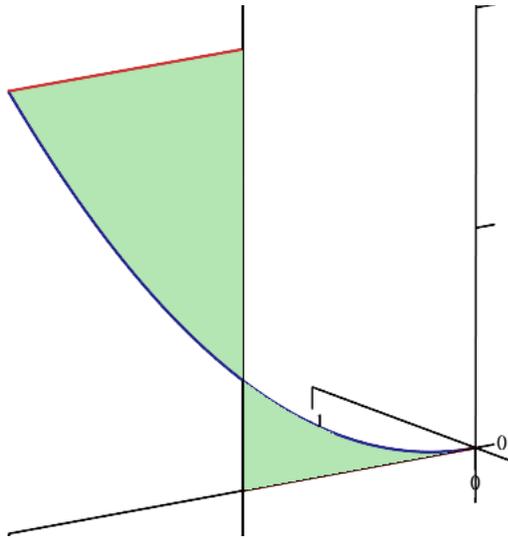
$$g2 := x \mapsto 4$$

(2.3.8.1)

```
> R12d:=plot2dy([f1(x),g1(x)],x=0..1,x=1):
R22d:=plot2dy([f2(x),g2(x)],x=1..2,x=1):
display(R12d,R22d);
```



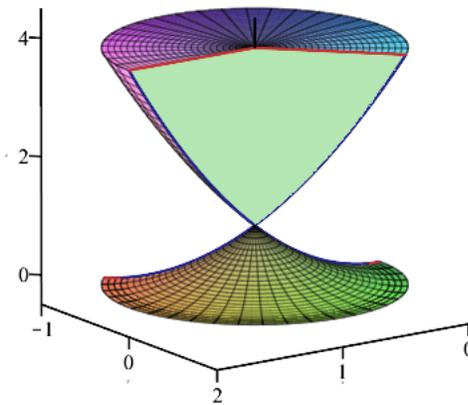
```
> R13d:=plot3dy([f1(x),g1(x)],x=0..1,x=1):
R23d:=plot3dy([f2(x),g2(x)],x=1..2,x=1):
display(R13d,R23d,orientation=[55,75],view=[0..2,-1..1,-0.5.
.4.5]);
```



```

> V1:=plotroty([f1(x),g1(x)],x=0..1,x=1,angle=-5*Pi/4,solide=
  fermé):
V2:=plotroty([f2(x),g2(x)],x=1..2,x=1,angle=-5*Pi/4,solide=
  fermé):
display(V1,V2,orientation=[55,75],view=[0..2,-1..1,-0.5..4.5]);

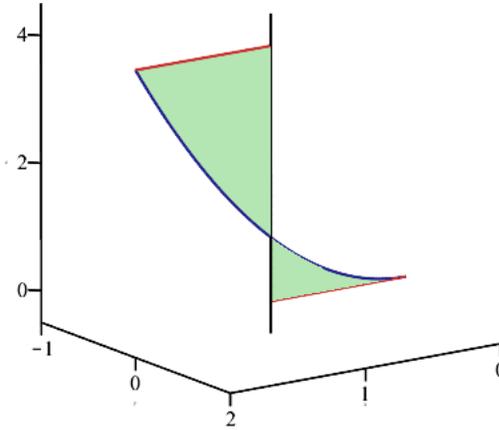
```



```

> A1:=plotrotyanim([f1(x),g1(x)],x=0..1,x=1,nimages=24,solide=
  fermé):
A2:=plotrotyanim([f2(x),g2(x)],x=1..2,x=1,nimages=24,solide=
  fermé):
display(A1,A2,orientation=[55,75],view=[0..2,-1..1,-0.5..4.5]);

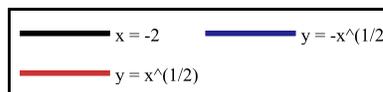
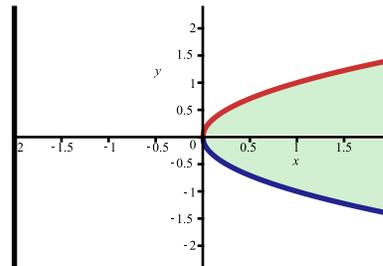
```



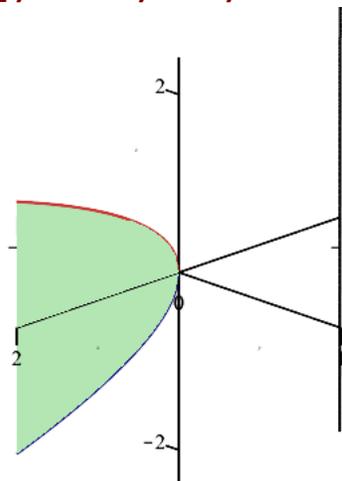
### Exemple 9

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = -\sqrt{x}$  et la courbe d'équation  $y = \sqrt{x}$  limitée par l'intervalle  $[0, 2]$ . L'axe de rotation est la droite d'équation  $x = -2$ .

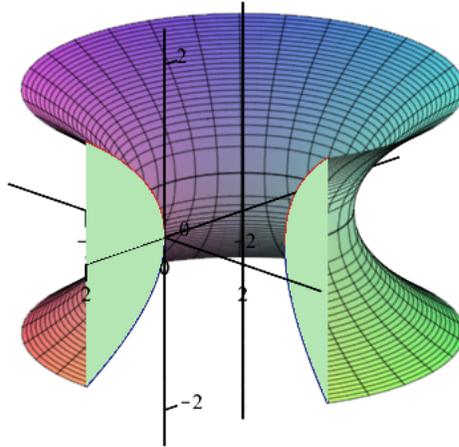
```
> f:=x->sqrt(x):
   plot2dy([-f(x),f(x)],x=0..2,x=-2);
```



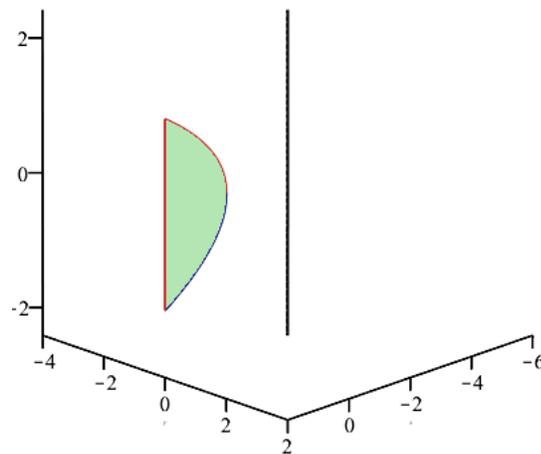
```
> plot3dy([-f(x),f(x)],x=0..2,x=-2,orientation=[45,70]);
```



```
> plotroty([-f(x),f(x)],x=0..2,x=-2,angle=-13*Pi/8,solide=ouvert,
orientation=[45,70],axes=normal);
```



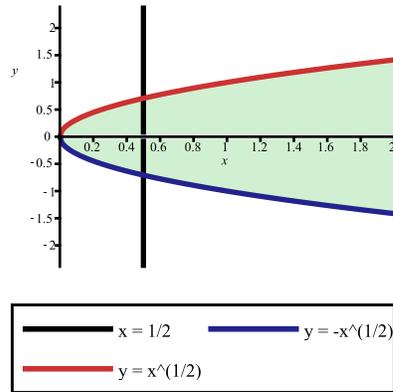
```
> A:=plotrotyanim([-f(x),f(x)],x=0..2,x=-2,nimages=-24,solid=
ouvert):
B:=plotcoqanim(x=-2,rayon=4,z=-sqrt(2)..sqrt(2),nimages=-24,
position=droite):
display(A,B,orientation=[45,70],axes=framed);
```



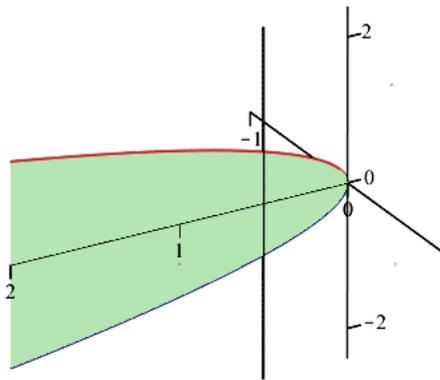
### Exemple 10

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = -\sqrt{x}$  et la courbe d'équation  $y = \sqrt{x}$  limitée par l'intervalle  $[0, 2]$ . L'axe de rotation est la droite d'équation  $x = \frac{1}{2}$ .

```
> f:=x->sqrt(x):
plot2dy([-f(x),f(x)],x=0..2,x=1/2);
```



```
> plot3dy([-f(x),f(x)],x=0..2,x=1/2,numpoints=400,orientation=[60,65]);
```

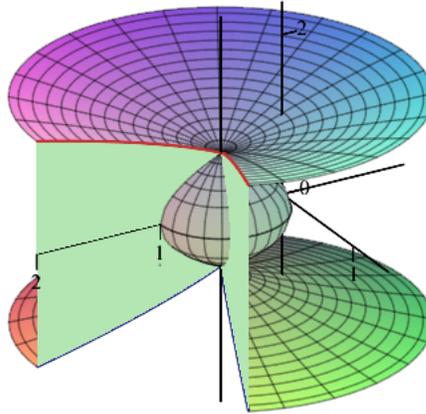


```
> plotroty([-f(x),f(x)],x=0..2,x=1/2,angle=-13*Pi/8,solide=ouvert,grid=[60,60],numpoints=200,orientation=[60,65],axes=normal);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

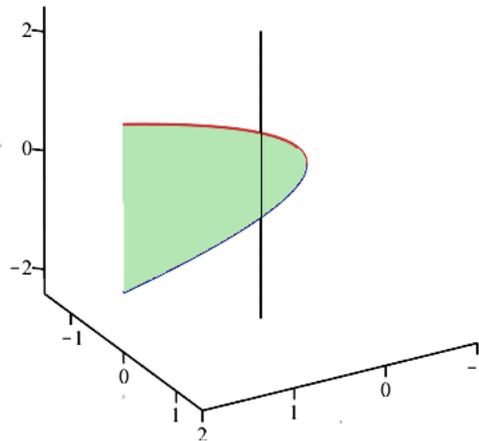


```
> plotrotyanim([-f(x),f(x)],x=0..2,x=1/2,nimages=-24,solid=ouvert,
grid=[60,60],numpoints=200,orientation=[60,65]);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

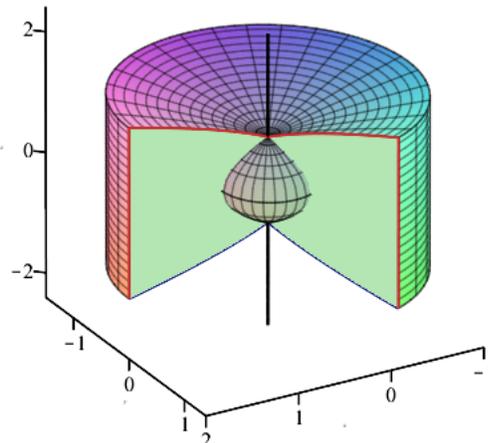


```
> A:=plotroty([-f(x),f(x)],x=0..2,x=1/2,angle=-11*Pi/8,solide=
ouvert,grid=[60,60],numpoints=200):
B:=plotcoq(x=1/2,rayon=3/2,z=-f(2)..f(2),angle=-11*Pi/8,
position=droite,grid=[60,60],numpoints=200):
display(A,B,orientation=[60,65]);
```

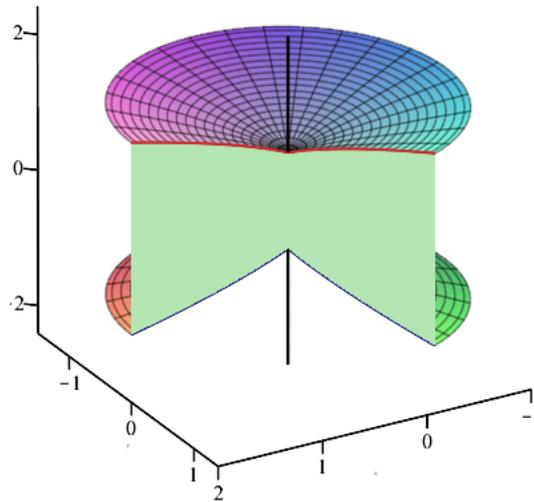
*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

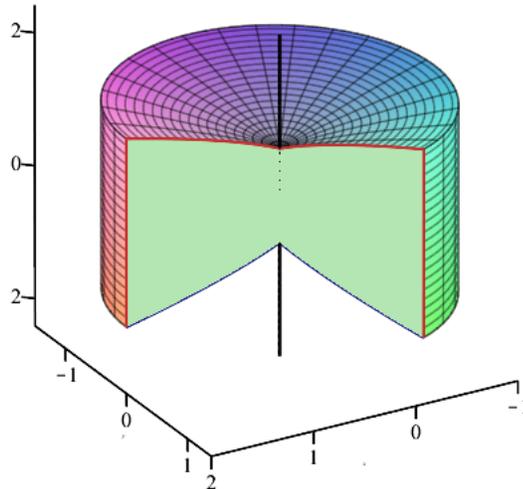
*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*



```
> A:=plotroty([-f(x),f(x)],x=1/2..2,x=1/2,angle=-11*Pi/8,solide=
ouvert,grid=[60,60],numpoints=200):
display(A,orientation=[60,65]);
```



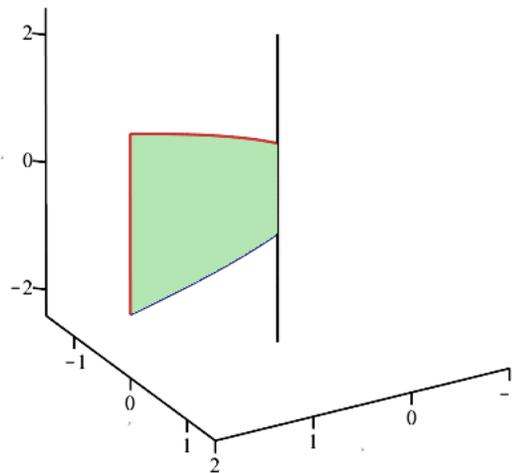
```
> B:=plotcoq(x=1/2,rayon=3/2,z=-f(2)..f(2),angle=-11*Pi/8,
position=droite,grid=[60,60],numpoints=200):
display(A,B,orientation=[60,65]);
```



```

> A:=plotrotyanim([-f(x),f(x)],x=1/2..2,x=1/2,nimages=-24,solid=
ouvert,grid=[60,60],numpoints=200):
B:=plotcoqanim(x=1/2,rayon=3/2,z=-f(2)..f(2),nimages=-24,
position=droite,grid=[60,60],numpoints=200):
display(A,B,orientation=[60,65]);

```



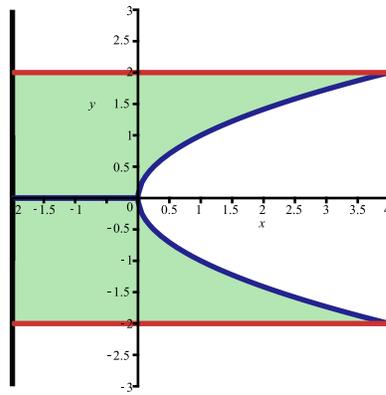
### Exemple 11

Nous pouvons saisir la région demandée avec une fonction continue par morceaux en utilisant la macro-commande *piecewise*, le résultat sera aussi bon que celui qui serait obtenu par la superposition de solides plus élémentaires.

```

> f:=x->piecewise(x>=-2 and x<=0,0,x>0 and x<=4,sqrt(x),infinity):
g:=x->2:
R12d:=plot2dy([f(x),g(x)],x=-2..4,x=-2):
R22d:=plot2dy([-f(x),-g(x)],x=-2..4,x=-2):
display(R12d,R22d,legend=["", ""]);

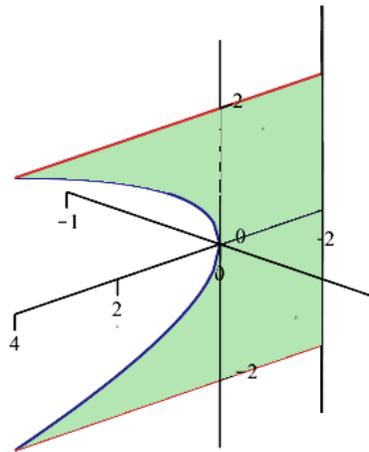
```



```

> R13d:=plot3dy([f(x),g(x)],x=-2..4,x=-2):
R23d:=plot3dy([-f(x),-g(x)],x=-2..4,x=-2):
display(R13d,R23d,axes=normal,orientation=[45,70]);

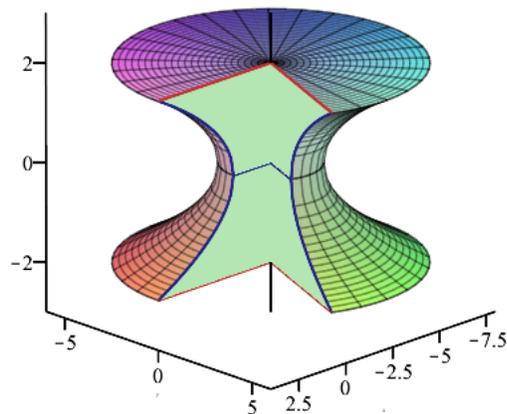
```



```

> V1:=plotroty([f(x),g(x)],x=-2..4,x=-2,angle=-13*Pi/8,solide=
fermé):
V2:=plotroty([-f(x),-g(x)],x=-2..4,x=-2,angle=-13*Pi/8,solide=
fermé):
display(V1,V2,orientation=[45,70]);

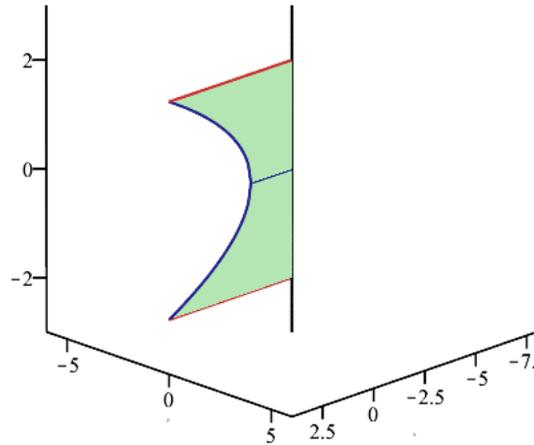
```



```

> A1:=plotrotyanim([f(x),g(x)],x=-2..4,x=-2,nimages=-24,solide=
fermé):
A2:=plotrotyanim([-f(x),-g(x)],x=-2..4,x=-2,nimages=-24,solide=
fermé):
display(A1,A2,orientation=[45,70]);

```



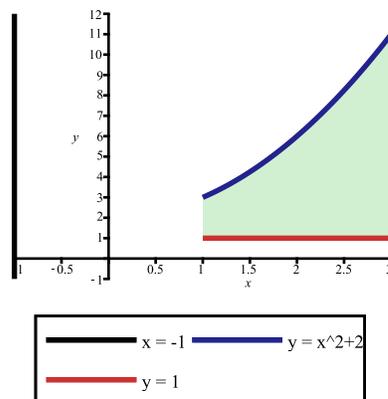
### Exemple 12

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = x^2 + 2$  et la courbe d'équation  $y = 1$  limitée par l'intervalle  $[1, 3]$ . L'axe de rotation est la droite d'équation  $x = -1$ .

```

> f:=x->x^2+2:
g:=x->1:
plot2dy([f(x),g(x)],x=1..3,x=-1);

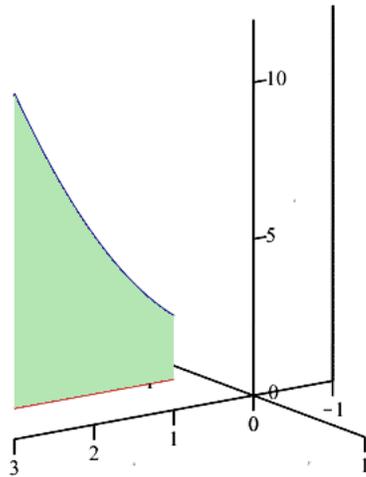
```



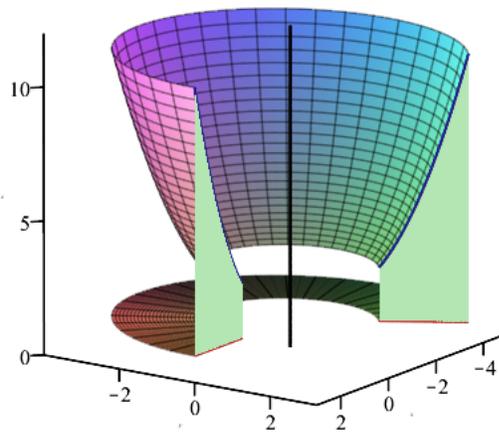
```

> plot3dy([f(x),g(x)],x=1..3,x=-1,axes=normal);

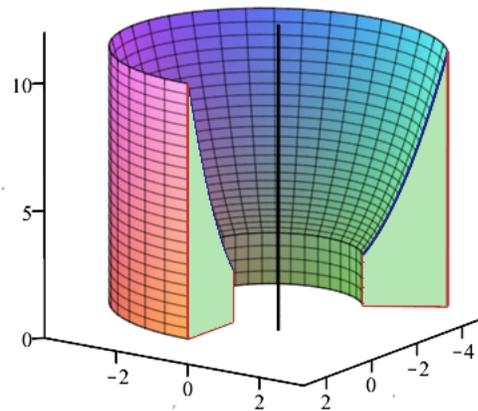
```



```
> A:=plototy([f(x),g(x)],x=1..3,x=-1,angle=-13*Pi/10,solide=
fermé):
display(A,orientation=[35,75]);
```



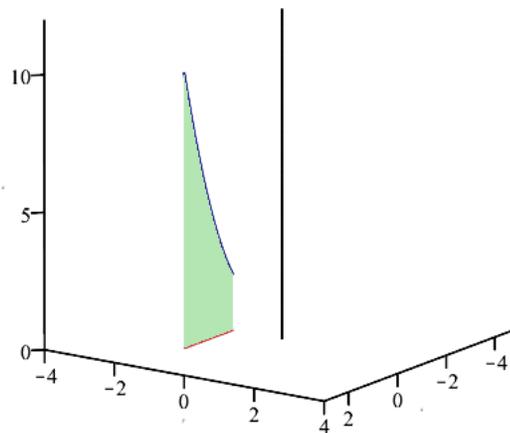
```
> B:=plotcoq(x=-1,rayon=4,z=f(3)..g(3),angle=-13*Pi/10,position=
droite):
C:=plotcoq(x=-1,rayon=2,z=f(1)..g(1),angle=-13*Pi/10,position=
droite,grid=[5,49]):
display(A,B,C,orientation=[35,75]);
```



```

> A1:=plotrotyanim([f(x),g(x)],x=1..3,x=-1,nimages=-24,solide=
ouvert):
A2:=plotcoqanim(x=-1,rayon=4,z=f(3)..g(3),nimages=-24,position=
droite):
A3:=plotcoqanim(x=-1,rayon=2,z=f(1)..g(1),nimages=-24,position=
droite,grid=[5,49]):
> display(A1,insequence=true,orientation=[35,75]);

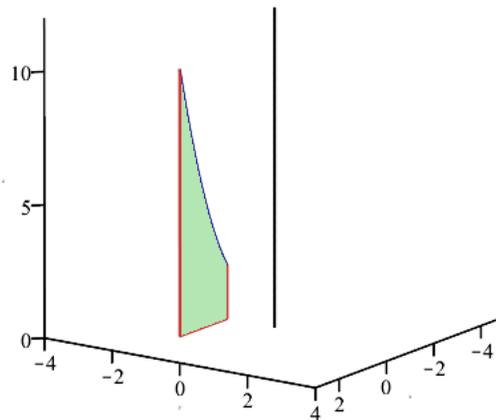
```



```

> display(A1,A2,A3,orientation=[35,75]);

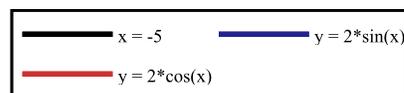
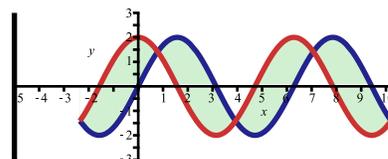
```



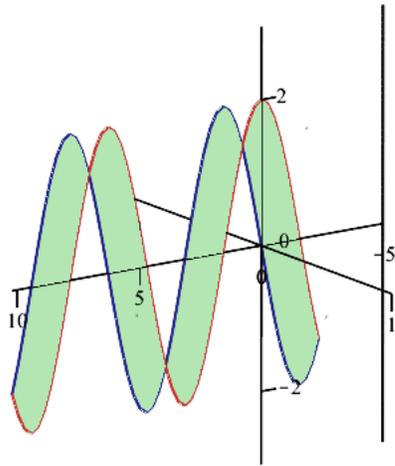
### Exemple 13

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = 2 \sin(x)$  et la courbe d'équation  $y = 2 \cos(x)$  limitée par l'intervalle  $\left[-\frac{3\pi}{4}, \frac{3\pi}{4}\right]$ . L'axe de rotation est la droite d'équation  $x = -5$ .

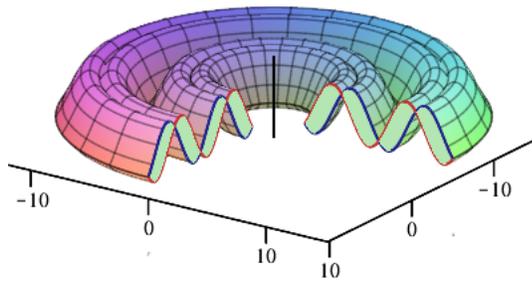
```
> f:=x->2*sin(x):
   g:=x->2*cos(x):
   plot2dy([f(x),g(x)],x=-3*Pi/4..13*Pi/4,x=-5,scaling=constrained)
   ;
```



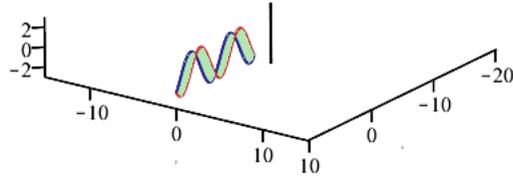
```
> plot3dy([f(x),g(x)],x=-3*Pi/4..13*Pi/4,x=-5);
```



```
> plototy([f(x),g(x)],x=-3*Pi/4..13*Pi/4,x=-5,angle=-3*Pi/2,
solide=ouvert,grid=[80,80],scaling=constrained,orientation=[35,
70]);
```



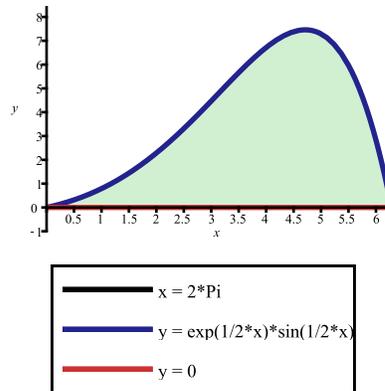
```
> plototyanim([f(x),g(x)],x=-3*Pi/4..13*Pi/4,x=-5,nimages=-24,
solide=ouvert,grid=[80,80],scaling=constrained,orientation=[35,
70]);
```



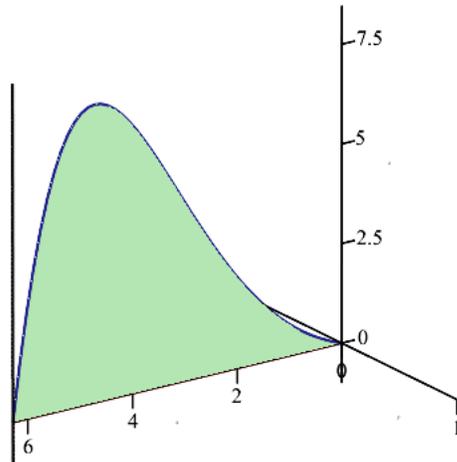
### Exemple 14

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y = e^{\frac{x}{2}} \sin\left(\frac{x}{2}\right)$  et l'axe des abscisses limitée par l'intervalle  $[0, 2\pi]$ . L'axe de rotation est la droite d'équation  $x = 2\pi$ .

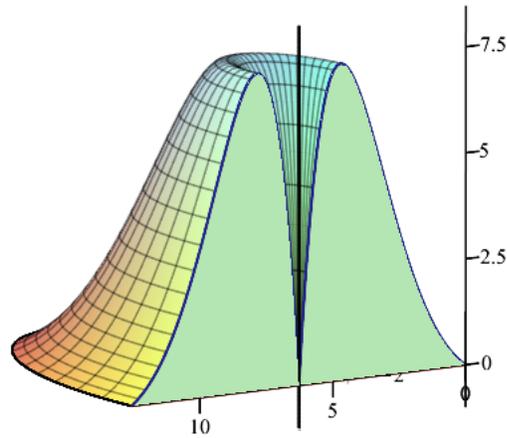
```
> f:=x->exp(x/2)*sin(x/2):
g:=x->0:
plot2dy([f(x),g(x)],x=0..2*Pi,x=2*Pi);
```



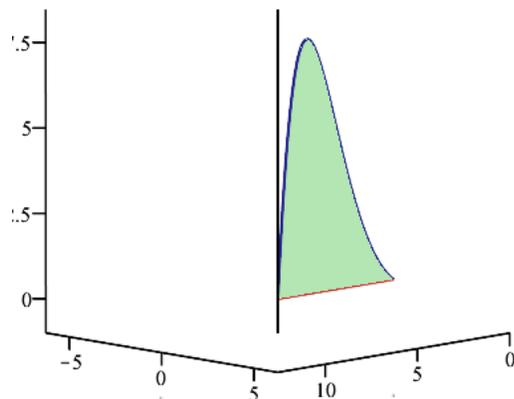
```
> plot3dy([f(x),g(x)],x=0..2*Pi,x=2*Pi,orientation=[55,70]);
```



```
> plotroty([f(x),g(x)],x=0..2*Pi,x=2*Pi,angle=Pi,slide=ouvert,
axes=normal,orientation=[55,80]);
```



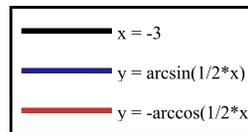
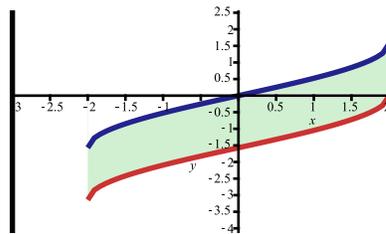
```
> plotrotyanim([f(x),g(x)],x=0..2*Pi,x=2*Pi,nimages=24,slide=
ouvert,orientation=[45,80]);
```



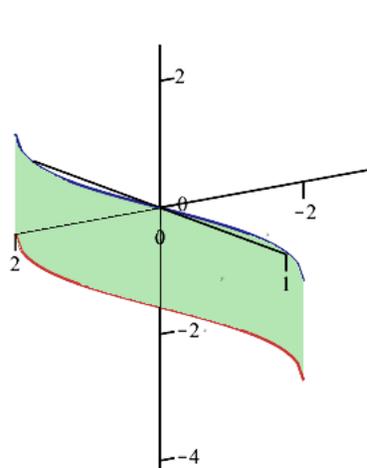
### Exemple 15

Soit le solide de révolution obtenu par la rotation de la surface comprise entre la courbe d'équation  $y_1 = \arcsin\left(\frac{x}{2}\right)$  et la courbe d'équation  $y_2 = -\arccos\left(\frac{x}{2}\right)$  limitée par l'intervalle  $[-2, 2]$ . L'axe de rotation est la droite d'équation  $x = -3$ .

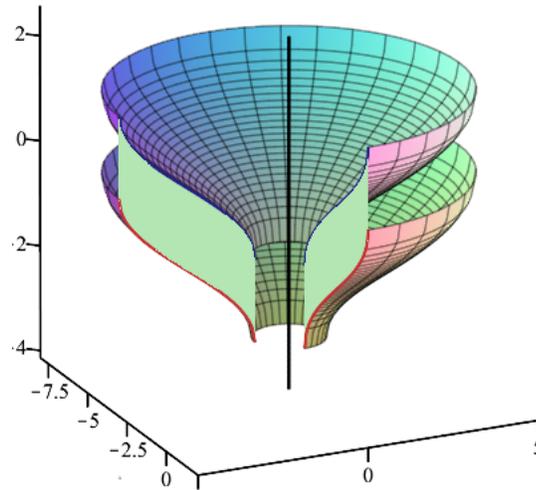
```
> f:=x->arcsin(x/2):  
g:=x->-arccos(x/2):  
plot2dy([f(x),g(x)],x=-2..2,x=-3);
```



```
> plot3dy([f(x),g(x)],x=-2..2,x=-3);
```



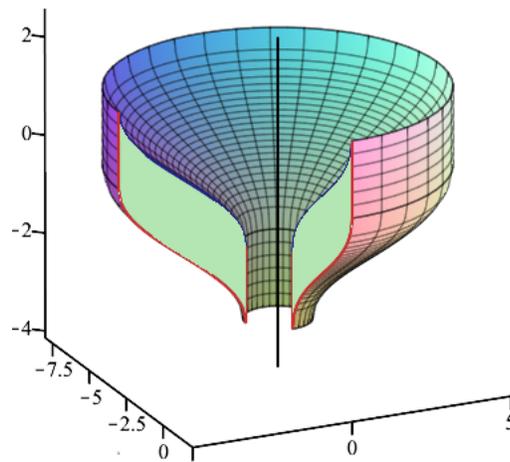
```
> Solide:=plotroty([f(x),g(x)],x=-2..2,x=-3,angle=3*Pi/2,solide=  
ouvert):  
display(Solide,orientation=[-25,70]);
```



```

> A:=plotcoq(x=-3,rayon=5,z=0..f(2),angle=3*Pi/2,position=droite,
grid=[7,49]):
B:=plotcoq(x=-3,rayon=1,z=g(-2)..f(-2),angle=3*Pi/2,position=
droite,grid=[7,49]):
display([Solide,A,B],orientation=[-25,70]);

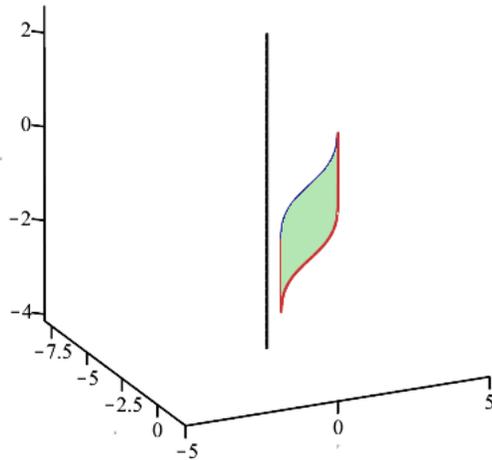
```



```

> Animation:=plotrotyanim([f(x),g(x)],x=-2..2.,x=-3,nimages=24,
solide=fermé):
Coq1:=plotcoqanim(x=-3,rayon=5,z=0..f(2),nimages=24,position=
droite,grid=[7,49]):
Coq2:=plotcoqanim(x=-3,rayon=1,z=g(-2)..f(-2),nimages=24,
position=droite,grid=[7,49]):
> display([Animation,Coq1,Coq2],orientation=[-25,70]);

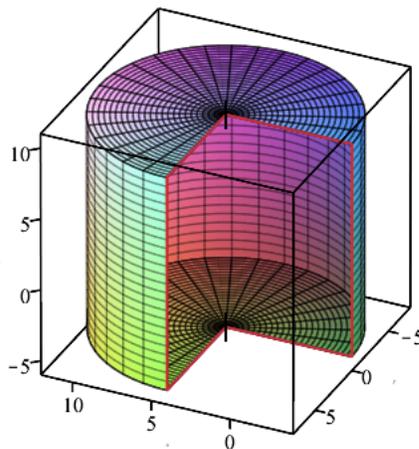
```



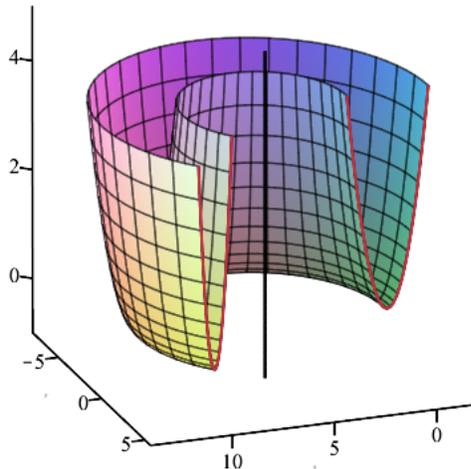
### Explorations: surfaces de rotation

Lorsque  $f(x) = g(x)$ , qu'est-ce qui se trace ?

```
> A:=plotroty([10,10],x=-4..4,x=4,angle=3*Pi/2,solide=fermé):
  B:=plotroty([-5,-5],x=-4..4,x=4,angle=3*Pi/2,solide=fermé):
  C:=plotcoq(x=4,rayon=8,z=-5..10,angle=3*Pi/2,position=gauche):
> display(A,B,C,orientation=[115,60],axes=boxed);
```



```
> plotroty([x^2,x^2],x=-2..2,x=6,angle=5*Pi/4,solide=fermé,
orientation=[70,70]);
```

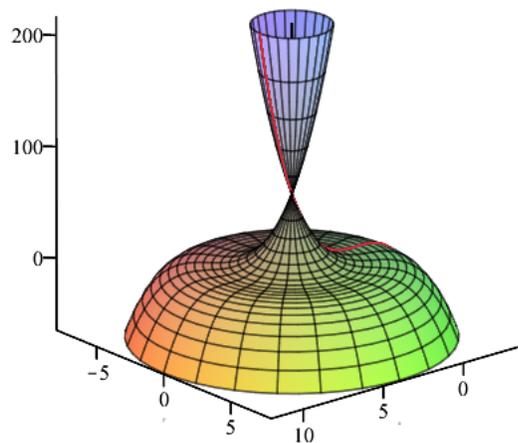


```
> plotroty([x^3,x^3],x=-4..6,x=4,angle=2*Pi,solide=ouvert,scaling=
unconstrained,orientation=[50,70]);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

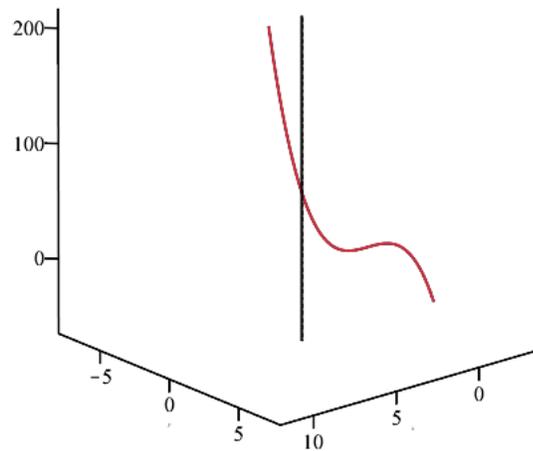


```
> plotrotyanim([x^3,x^3],x=-4..6,x=4,nimages=24,solide=ouvert,
grid=[12,25],scaling=unconstrained,orientation=[50,70]);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*



```
> plotrotanim([x^2,x^2],x=-4..6,x=4,nimages=24,solide=ouvert,
scaling=unconstrained,orientation=[50,70]);
```

*Remarque:*

*L'axe de révolution chevauche la surface. Il est donc possible d'obtenir un même solide par superposition*

*de sous-régions efficaces où l'axe de révolution ne chevauchera aucune de ces sous régions.*

