



Solides de révolution (Première partie de trois)

© Pierre Lantagne

Enseignant retraité du Collège de Maisonneuve

Ce document est une révision de celui produit en 2005. L'objectif principal de ce premier document est d'initier le lecteur aux tracés de courbes et de surfaces dans l'espace. Par soucis pédagogique, les surfaces qui seront tracées dans l'espace correspondront à la visualisation faite en classe de surfaces engendrées par la rotation de courbes autour de l'axe des x ou de l'axe des y du plan cartésien. Les deux autres documents de cette série permettront au lecteur d'être introduit à un ensemble de procédures dédiées aux tracés de solides de révolution. Ces procédures, écrites pour mes élèves, ont été réunies dans une bibliothèque que j'ai intitulée *riemann*.

Bonne lecture à tous !

* Ce document Maple est exécutable avec la version 2020.2

Initialisation

```
> restart;
> macro(grispâle = COLOR(RGB, .9253, .9253, .9253)):
with(plots,animate,display,setoptions,setoptions3d,spacecurve);
setoptions(size=[400,400],labels=[x,y],tickmarks=[12,12],axesfont=
[TIMES,ROMAN,8],labelfont=[TIMES,ROMAN,8]);
setoptions3d(size=[400,400],labels=[x,y,z],axes=normal,tickmarks=[4,
4,4],axesfont=[TIMES,ROMAN,8],labelfont=[TIMES,ROMAN,8]):
[animate,display,setoptions,setoptions3d,spacecurve] (1.1)
```

...Vers la troisième dimension

Rotation de $y = x^2$ autour de l'axe des y

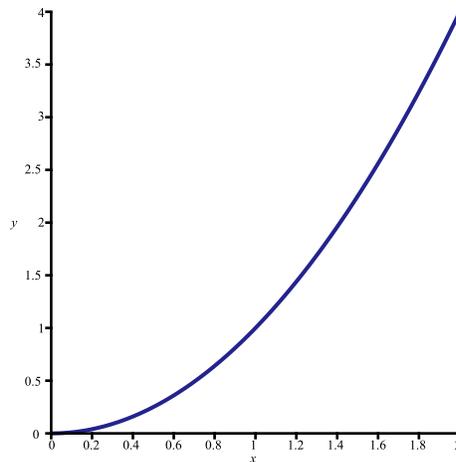
Dans l'ensemble de mes documents Maple, la syntaxe utilisée pour le tracé des graphiques à deux dimensions est la syntaxe paramétrique. La plupart des tracés que mes élèves ont eu à effectuer dans leurs travaux Maple ont toujours été pour des courbes définies explicitement par une relation de la forme $y = f(x)$. La syntaxe paramétrique est donc de la forme `plot([x,f(x),x=a..b], options)`.

L'utilisation de la syntaxe paramétrique dans les tracés dans le plan va donc faciliter le passage vers des tracés de courbes ou de surfaces dans l'espace.

Commençons avec le tracé suivant dans le plan cartésien.

Soit la fonction f définie par $f(x) = x^2$. Saisissons la fonction f avec l'opérateur flèche et traçons son graphe sur l'intervalle $[0, 2]$.

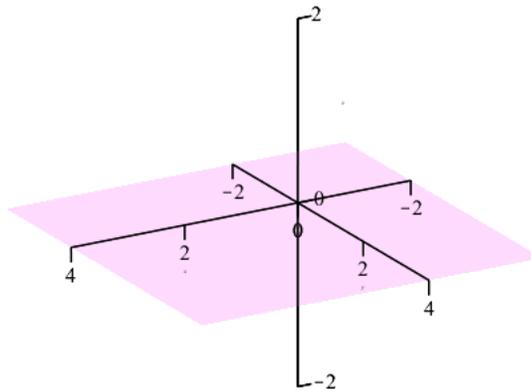
```
> f:=x->x^2;
plot([x,f(x),x=0..2],color=navy);
f := x ↦ x2
```



Pour visualiser la surface engendrée par la **rotation de ce tracé autour de l'axe des y**, nous allons d'abord obtenir ce tracé dans un plan de l'espace.

Obtenons ce tracé dans le plan xOy . Visualisons ce plan à l'aide de la macro-commande `plot3d`.

```
> Plan_xOy:=plot3d([s,t,0],s=-2..4,t=-2..4,color=plum,transparency=
0.25,style=patchnogrid):
display(Plan_xOy,view=[-2..4,-2..4,-2..2],orientation=[60,70]);
```



Avec un clic-gauche tout en maintenant enfoncé le bouton de la souris, sélectionnez le graphique précédent. Glissez ensuite la souris afin de modifier l'orientation du graphique pour percevoir pleinement la troisième dimension.

La macro-commande `spacecurve` de la bibliothèque `plots` permet de transposer facilement dans l'espace un tracé du plan cartésien.

Saisissons d'abord les fonctions g et h qui serviront à définir, respectivement, l'abscisse et l'ordonnée des points de l'espace afin de transposer le tracé dans le plan $y=0$.

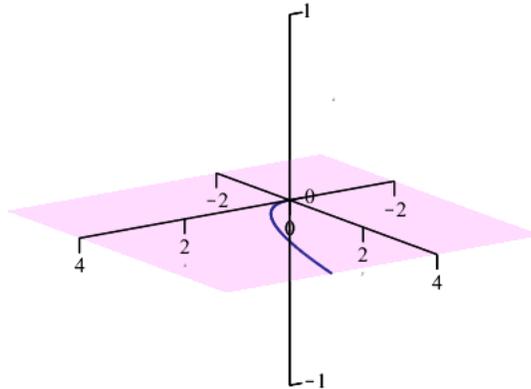
```
> g:=s->s;
h:=s->s^2;
```

$$g := s \mapsto s$$

$$h := s \mapsto s^2$$

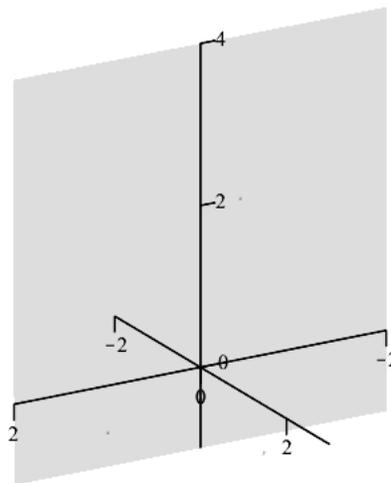
(2.1.1)

```
> Courbe3d:=spacecurve([g(s),h(s),0,s=0..2],thickness=2,color=navy):
display(Plan_xOy,Courbe3d,view=[-2..4,-2..4,-1..1],axes=normal);
```

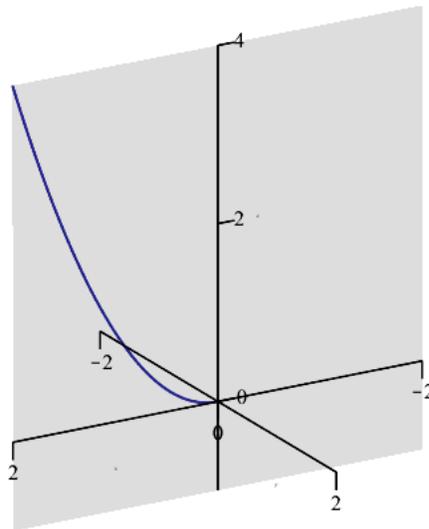


Puisqu'en classe, la représentation de la rotation autour de l'axe des y est faite à partir du graphique dans un plan cartésien et donc que la surface de rotation est représentée verticalement, par soucis pédagogique, le tracé sera plutôt effectué dans le plan xOz de l'espace, c'est-à-dire le plan d'équation $y=0$. Nous allons effectuer cette rotation dans l'espace cartésien où l'axe des z sera l'axe de rotation autour duquel pivotera le tracé.

```
> Plan_xOz:=plot3d([s,0,t],s=-2..2,t=-1..4,color=
grispâle,transparency=0.75,style=patchnogrid):
display(Plan_xOz,view=[-2..2,-2..3,-1..4],orientation=[60,70]);
```



```
> Courbe3d:=spacecurve([g(s),0,h(s),s=0..2],thickness=2,color=navy):
display(Plan_xOz,Courbe3d,view=[-2..2,-2..2,-1..4],orientation=
[60,70]);
```



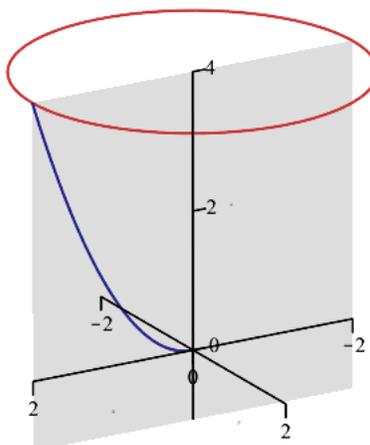
Créons, dans le plan $z = 4$, un cercle de rayon 2 centré au point $[0, 0, 4]$ et superposons-le au tracé précédent. Cette construction graphique est, somme toute, assez simple. Pour chaque point (x, y, z) de ce cercle, les coordonnées paramétriques sont

$$\begin{aligned} x &= g(s) \cos(t) \\ y &= g(s) \sin(t) \\ z &= h(s) \end{aligned}$$

En effet, pour chaque valeur réelle de la variable s , lorsque t variera de 0 à 2π , on obtiendra le tracé d'une cercle de rayon $g(s)$ dans le plan de l'espace d'équation $z = h(s)$, centré au point $(0, 0, h(s))$.

Posons $s = 2$ pour le tracé du cercle à faire dans le plan $z = 4$. Réalisons maintenant la superposition demandée.

```
> Cercle3d:=spacecurve([g(2)*cos(t),g(2)*sin(t),h(2)],t=0..2*Pi,
                        color=orange,thickness=2):
Lieux:=Cercle3d,Courbe3d:
display(Plan_xOz,Lieux,view=[-2..2,-2..2,-1..4],orientation=[60,
70]);
```

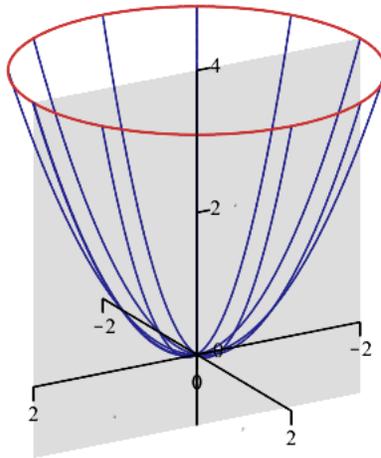


Visualisons maintenant la rotation de cette portion de courbe autour de l'axe des z en superposant onze

copies du tracé de telle façon que la $k^{\text{ème}}$ copie subisse une rotation d'un angle $\theta = \frac{k\pi}{6}$.

Superposons, dans un même graphique, toutes ces copies et le tracé original de la portion en couleur navy.

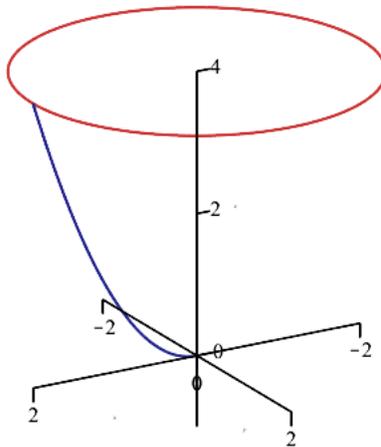
```
> Copies:=seq(spacecurve([g(s)*cos(k*Pi/6),g(s)*sin(k*Pi/6), h(s)],
s=0..2,
                    thickness=1,color=navy),k=1..11):
display([Plan_xOz,Copies,Lieux],view=[-2..2,-2..2,-1..4],
orientation=[60,70]);
```



Afin d'accentuer l'effet de rotation, nous allons animer les tracés des copies de la courbe originale en les affichant séquentiellement. Nous allons de plus superposer quatre cercles dans les plans d'équations $z = 0.5$, $z = 1$, $z = 2$ et $z = 3$.

La création des fonctions *Rotation*, *Cercle_z* et *Révolution_z* nous permettra d'automatiser cette animation.

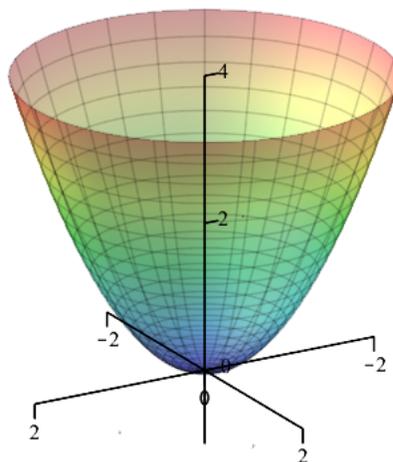
```
> Rotation:=(theta,Couleur)->spacecurve([g(s)*cos(theta),g(s)*sin
(theta),h(s),s=0..2],
                    thickness=2,color=Couleur):
Cercle_z:=(Niveau,Pas,Couleur)->spacecurve([g(Niveau)*cos(t),g
(Niveau)*sin(t),h(Niveau)],
                    t=0..Pas,thickness=1,color=
Couleur):
n:=24: # Nombre de séquences pour l'animation
Révolution_z:=z->display(display(Lieux),
                        [seq(display([Rotation(Pi/12*k,navy),
seq(Cercle_z(sqrt(1),Pi/12*k,orange),l=[.5,1,2,3]
),
                        Cercle_z(sqrt(z),Pi/12*k,magenta),Lieux]),
k=1..n)],insequence=true):
> Animation:=Révolution_z(sqrt(2)):
> display(Animation,insequence=true,view=[-2..2,-2..2,-1..4],
orientation=[60,70]);
```



Sélectionnez le graphique précédent et, dans la barre contextuelle du menu, cliquez sur le bouton de départ pour lancer l'animation.

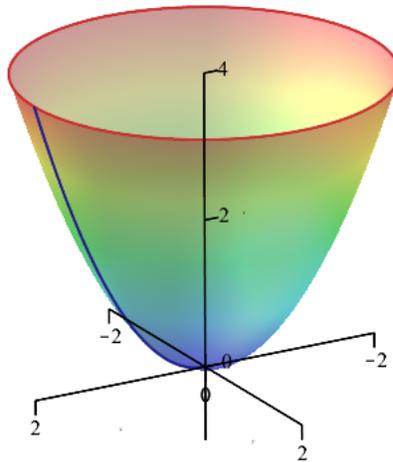
La macro-commande *plot3d* de la bibliothèque de base permet d'obtenir directement le tracé de la surface engendré par la rotation de cette portion de courbe.

```
> Parabolöide:=plot3d([g(s)*cos(t),g(s)*sin(t),h(s)],s=0..2,t=0..2*
Pi,transparency=0.6):
display(Parabolöide,shading=zhue,view=[-2..2,-2..2,-1..4],
orientation=[60,70]);
```



Superposons l'animation précédente avec la surface ci-dessus. Choisissons le style de surface *patchnograd*.

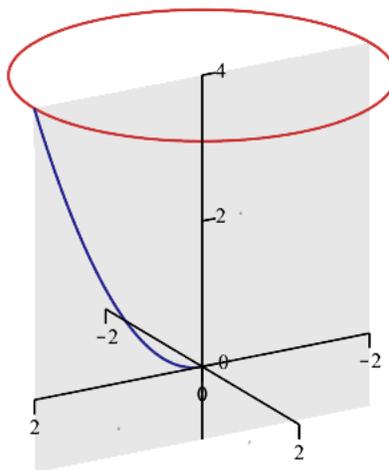
```
> display([Parabolöide,Animation],shading=zhue,style=patchnograd,
view=[-2..2,-2..2,-1..4],orientation=[60,70]);
```



Il s'agit toujours d'une animation. Sélectionnez le graphique et, dans la barre contextuelle du menu, cliquez sur le bouton de départ pour lancer l'animation.

Voici une autre animation illustrant cette surface de révolution.

```
> Fond:=display(Plan_xOz,Lieux):
animate( plot3d, [[g(s)*cos(t),g(s)*sin(t),h(s)],s=0..2,t=0..B],
          B=0..2*Pi,frames=24,background=Fond,style=patchnograd,
          lightmodel=light4,view=[-2..2,-2..2,-1..4],orientation=[60,70]);
          B=0.0000000000
```



Il s'agit encore d'une animation. Sélectionnez le graphique et, dans la barre contextuelle du menu, cliquez sur le bouton de départ pour lancer l'animation.

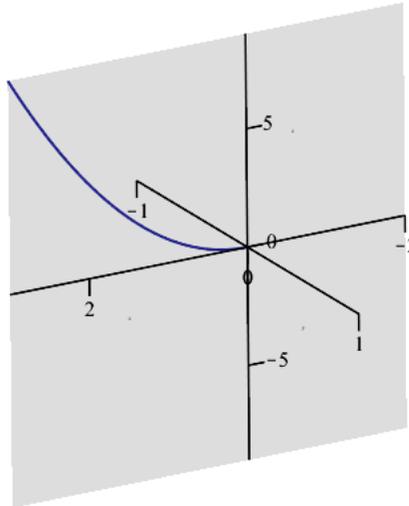
Rotation de $y=x^2$ autour de l'axe des x

Illustrons la rotation de la même courbe mais, cette fois-ci, autour de l'axe des x .

Affichons de nouveau la courbe dans l'espace pour visualiser la surface engendrée par la **révolution de cette courbe autour de l'axe des x** .

Pour une bonne illustration, considérons $x \in [0, 3]$ et effectuons un affichage sans échelle de contrainte.

```
> Plan_xOz:=plot3d([s,0,t],s=-2..3,t=-9..9,color=
grispâle,transparency=0.75,style=patchnogrid):
Courbe3d:=spacecurve([g(s),0,h(s),s=0..3],thickness=2,color=navy):
display(Plan_xOz,Courbe3d,orientation=[60,70]);
```



Afin de visualiser la rotation de la courbe d'équation $y = x^2$ autour de l'axe des x du plan cartésien, nous devons, ici également, réaliser cette rotation dans l'espace. L'axe des x sera l'axe de rotation autour duquel pivotera la courbe tracée dans l'espace.

Superposons un cercle de rayon 9 centré au point $(3, 0, 0)$ dans le plan d'équation $x = 3$. La construction à faire repose sur le même principe que précédemment: pour chaque point (x, y, z) de ce cercle, les coordonnées sont, cette fois,

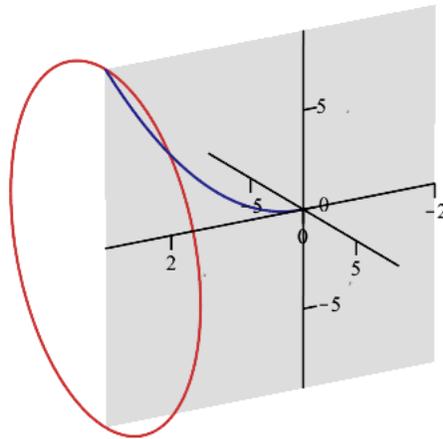
$$\begin{aligned}x &= g(s); \\ y &= h(s) \sin(t); \\ z &= h(s) \cos(t);\end{aligned}$$

En effet, pour chaque valeur de la variable s , lorsque t variera de 0 à 2π , on obtient le tracé d'un cercle de rayon $h(s)$ dans le plan de l'espace $x = g(s)$ centré au point $(g(s), 0, 0)$.

Posons $s = 3$ pour le tracé du cercle à faire dans le plan $x = 3$.

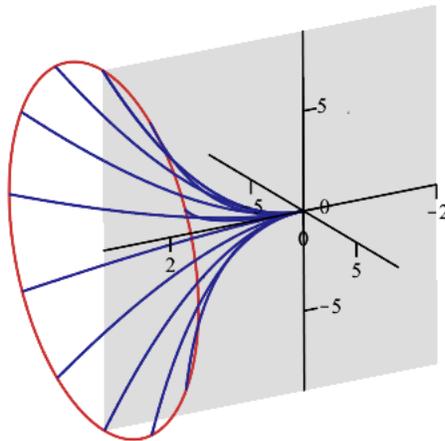
Réalisons maintenant la superposition demandée.

```
> Cercle3d:=spacecurve([g(3),h(3)*sin(t),h(3)*cos(t)],t=0..2*Pi,
color=orange,thickness=2):
Lieux:=Cercle3d,Courbe3d,Plan_xOz:
display(Lieux,orientation=[60,70]);
```



Visualisons maintenant la rotation de la courbe autour de l'axe des x en superposant onze copies de la courbe de telle façon que la $k^{\text{ième}}$ copie subit une révolution d'un angle $\theta = \frac{k\pi}{6}$. Superposons toutes ces copies et le tracé original.

```
> Copies:=seq(spacecurve([g(s),h(s)*cos(k*Pi/6),
                        h(s)*sin(k*Pi/6)],s=0..3,
                        thickness=2,color=navy),k=1..11):
display([Lieux,Copies],orientation=[60,70]);
```



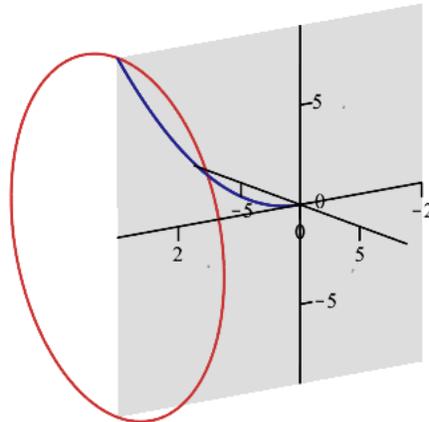
Afin d'augmenter l'effet de rotation, animons aussi les tracés des copies de la courbe originale en les affichant séquentiellement. Nous allons de plus superposer quatre cercles dans les plans d'équations $x = 0.5$, $x = 1$, $x = 2$ et $x = 3$. La création de la macro-commande **Révolution_x** nous permettra d'automatiser cette révolution.

```
> Rotation:=(theta,Couleur)->spacecurve(
        [g(s),h(s)*cos(theta+Pi/2),h(s)*sin(theta+Pi/2),s=0.
        .3],
        thickness=2,color=Couleur):
Cercle_x:=(Niveau,Pas,Couleur)->spacecurve(
```

```

    [g(Niveau),h(Niveau)*cos(t+Pi/2),h(Niveau)*sin(t+Pi/2)],
    t=0..Pas,thickness=1,color=Couleur):
n:=24: # Nombre de séquences pour l'animation
Révolution_x:=x->display([seq(display([Rotation(Pi/12*k,navy),
    seq(Cercle_x(l,Pi/12*k,orange),l=[0.5,1,1.5,2,2.5,3]
    ),Cercle_x(x,Pi/12*k,magenta),Lieux]),k=0..n)],insequence=true):
> Animation:=Révolution_x(1.5):
> display(Animation,insequence=true,orientation=[60,70]);

```

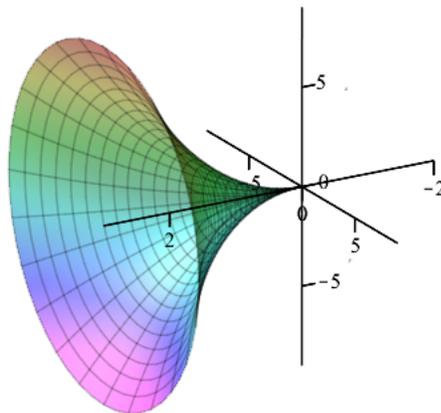


Traçons directement la surface de cette trompette à l'aide de la macro-commande *plot3d*.

```

> Trompette:=plot3d([g(s),h(s)*cos(t),h(s)*sin(t)],s=0..3,t=0..2*Pi,
shading=zhue,transparency=0.45):
display(Trompette,orientation=[60,70],view=[-2..3,-9..9,-9..9]);

```

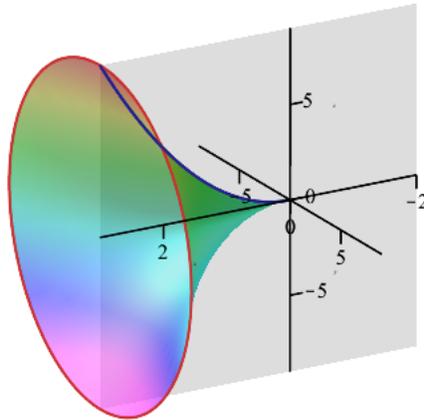


Pour terminer, superposons l'animation précédente avec la surface ci-dessous avec le style de surface *patchnograd*

```

> display([Trompette,Animation],shading=zhue,style=patchnograd,
orientation=[60,70]);

```



```

> Fond:=display(Lieux):
animate( plot3d, [[g(s),h(s)*cos(t),h(s)*sin(t)],s=0..3,t=Pi/2..B,
shading=zhue],
        B=Pi/2..5*Pi/2,frames=24,background=Fond,style=
patchnogrid,orientation=[60,70],lightmodel=light4,view=[-2..3,-9.
.9,-9..9]);

```

B = 1.5708000000

