

Exercice 1 _____ **Matrices**

1. Soit A une matrice carrée, que valent $\text{diag}(A)$ et $\text{diag}(\text{diag}(A))$?
2. Les fonctions `tril` (resp. `triu`) permettent d'extraire la partie triangulaire inférieure (resp. supérieure) d'une matrice. Définir une matrice carrée A quelconque (par exemple avec la fonction `rand()`) et construire en une seule instruction, une matrice triangulaire inférieure T telle que $T_{ij} = A_{ij}$ pour $i > j$ (les parties strictement triangulaires inférieures de A et T sont égales) et $T_{ii} = 1$ pour tout i (T est à diagonale unité).

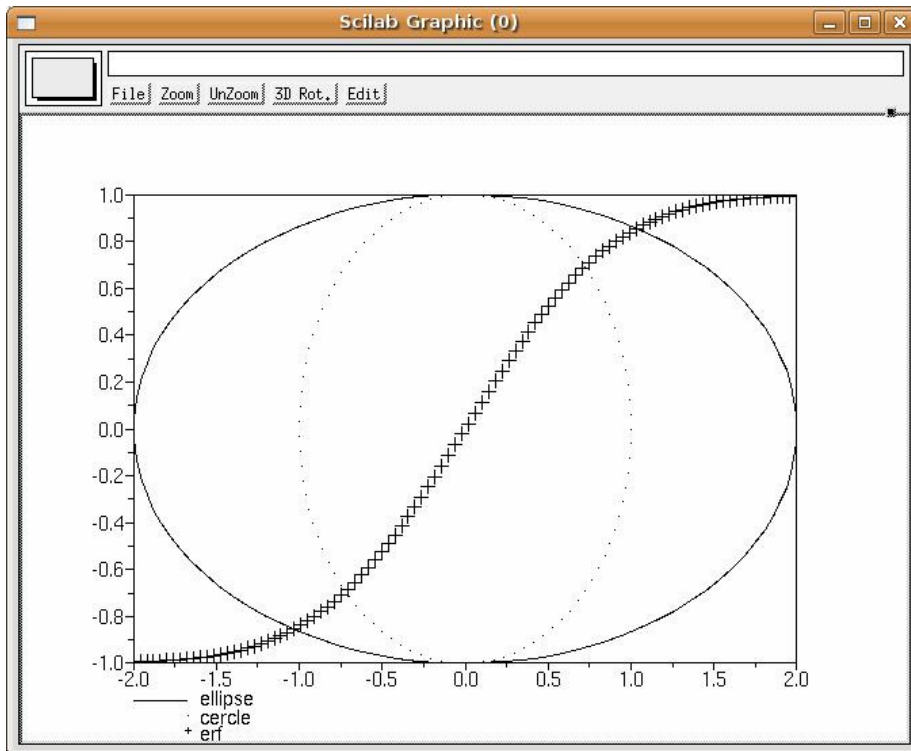
Exercice 2 _____ **Dessin 2D**

Tracer sur un même graphe les trois fonctions suivantes :

$$\begin{cases} x = 2 * \cos(t) \\ y = \sin(t) \end{cases} \quad t \in [0, 2\pi],$$

$$\begin{cases} x = \cos(t) \\ y = \sin(t) \end{cases} \quad t \in [0, 2\pi]$$

et la fonction `erf(x)` de Scilab, pour x variant de -2 à 2 . Utiliser l'instruction `xset("use color", 0)` pour passer en mode noir et blanc. Consulter l'aide en ligne de Scilab pour connaître les arguments optionnels de `plot2d` et présenter les résultats selon le modèle ci-dessous (axes gradués, un style différent et une légende pour chaque courbe).



Exercice 3 Fonctions

- Écrire la fonction `[x,y]=hyperbole(a,b,x0,p)` qui calcule les vecteurs `x` et `y` des coordonnées x et (respectivement) y des points de l'hyperbole $\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$ selon l'algorithme suivant :
 - calcule la matrice `X1Y1` dont les deux colonnes contiennent les coordonnées X_1 et Y_1 des points de la courbe $Y_1 = \frac{-a^2}{2X_1}$, X_1 variant de $-x_0$ à x_0 (**sans prendre la valeur nulle**) avec un pas de discrétisation `p`,
 - calcule la matrice `X2Y2 = X1Y1*R'` où `R` est la matrice de rotation d'angle $\frac{\pi}{4}$,
 - calcule `x` et `y` par affinité orthogonale d'axe (Ox) et de rapport `b/a` de l'hyperbole représentée par la matrice `X2Y2`.
- Tracer le graphe de cette fonction pour différentes valeurs des arguments.

Exercice 4 Dessin 3D

Les fichiers OFF (Object File Format) sont utilisés pour représenter la géométrie d'objets 3D sous forme de listes de polygones. Le format d'un fichier OFF est le suivant :

```
OFF
nbSommets nbFaces nbSegments
x y z
x y z
... (nbSommets lignes)
nbSommetsFace1 s1 s2 s3 ... sN
nbSommetsface2 s1 s2 s3 ... sM
... (nbFaces lignes)
```

où `s1`, `s2`, ... sont les indices des sommets par ordre d'apparition dans le fichier en commençant par 0. Par exemple pour un cube :

```
OFF 8 6 0
1 0 0
0 1 0
0 0 1
0 0 0
1 0 1
1 1 0
1 1 1
0 1 1
4 0 5 6 4
4 1 5 6 7
4 2 4 6 7
4 3 0 4 2
4 3 0 5 1
4 3 1 7 2
```

- Écrire la fonction `[s,f]=read_off(fname)` qui lit un fichier OFF de nom `fname` en utilisant les fonctions `mopen`, `mfscanf` et `mclose` et retourne :
 - la matrice `s` des sommets (une ligne de 3 coordonnées par sommet)
 - la matrice `f` des faces (une ligne d'indices de sommets par face) : les indices des sommets doivent ici commencer à 1 (comme les indices Scilab) et comme toutes les faces n'ont pas nécessairement le même nombre de sommets, certaines lignes peuvent être complétées par des 0.
- Écrire la fonction `plot_off(fname)` qui appelle la fonction `read_off` et dessine les faces du modèle OFF en utilisant la fonction `param3d` (remarque : tout comme avec `plot2d`, une discontinuité dans le tracé de `param3d` peut être obtenue en intercalant un point de coordonnées `(%nan,%nan,%nan)`). Appeler cette fonction avec les fichiers `cube.off`, `helix2.off` et `space_shuttle.off` disponibles à l'adresse <http://webloria.loria.fr/~penarand/bgi/>