

TP7 En Python

Formes Géométriques

Allumer l'ordinateur et connectez-vous en utilisant votre login et votre mot de passe puis lancer « Python en ligne : <https://www.codabrainy.com/python-compiler/> »

Activité

L'étude graphique d'objets géométriques à l'aide d'outils numériques permet de passer d'une figure abstraite à une représentation visuelle précise. En utilisant Python et des bibliothèques comme NumPy et Matplotlib, il devient possible de tracer des points régulièrement répartis sur un cercle, de comprendre leur organisation et de visualiser les formes qui en résultent. Ce type de démarche allie raisonnement mathématique et exploration graphique, offrant un terrain idéal pour expérimenter la géométrie sous une forme dynamique et moderne.

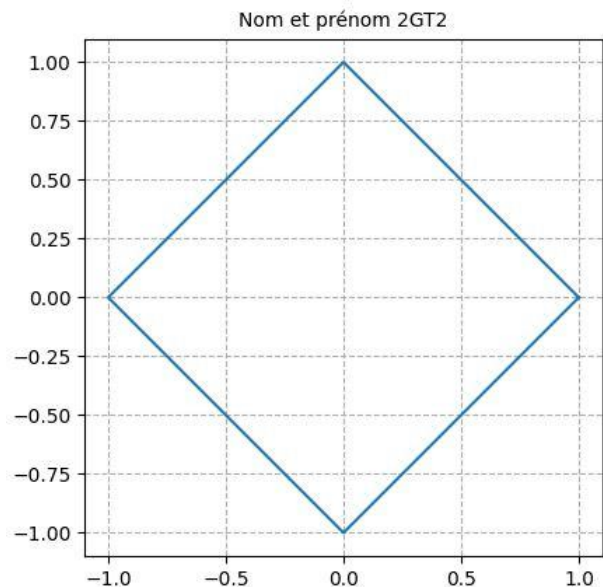
Activité N°1 (il trace un polygone fermé reliant cinq points régulièrement espacés sur le cercle unité)

```
import numpy as np
import matplotlib.pyplot as plt

plt.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 5)
r = np.sqrt(1.0)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
fig, ax = plt.subplots(1)
ax.plot(x1, x2)
ax.set_aspect(1)

plt.grid(linestyle='--')
plt.title('Nom et prénom 2GT18', fontsize=10)

plt.show()
```



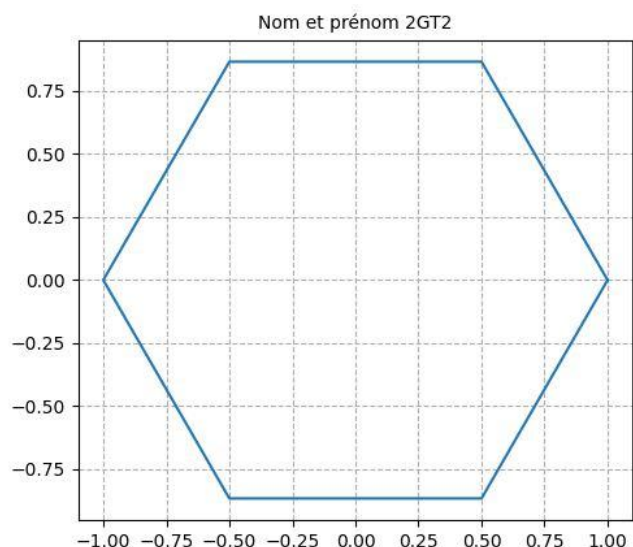
Activité N°2 (Script : trace un polygone régulier à six côtés, un petit hexagone inscrit dans le cercle unité.

```
import numpy as np
import matplotlib.pyplot as plt

plt.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 7)
r = np.sqrt(1.0)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
fig, ax = plt.subplots(1)
ax.plot(x1, x2)
ax.set_aspect(1)

plt.grid(linestyle='--')
plt.title('Nom et prénom 2GT18', fontsize=10)

plt.show()
```



Activité N°3 (Cette fois la figure devient un décagone, dix points étalés autour du cercle unité)

```
import numpy as np
import matplotlib.pyplot as plt

plt.axis([-1.25,1.25,-1.25,1.25])

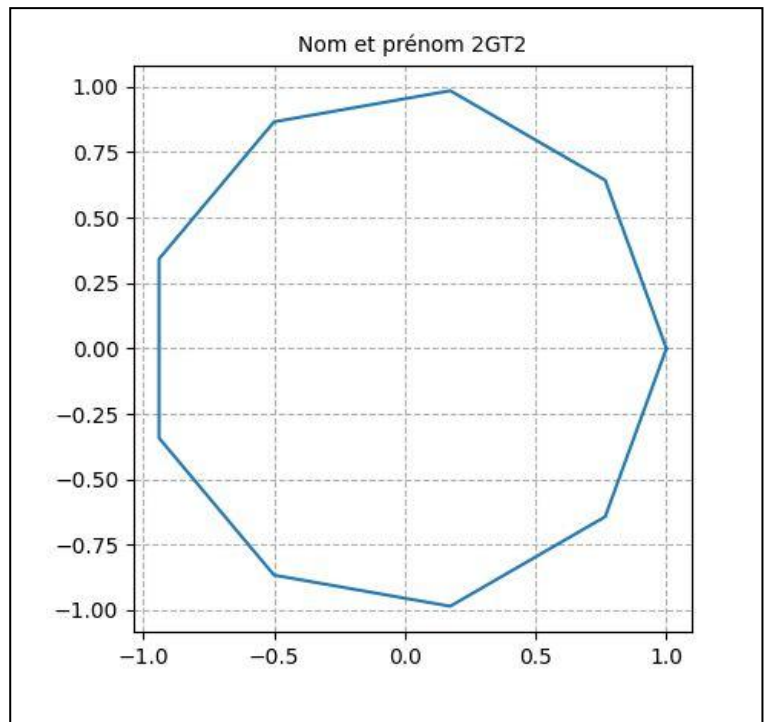
a = np.linspace(0, 2*np.pi, 10)
r = np.sqrt(1.0)
x1 = r*np.cos(a)
x2 = r*np.sin(a)

fig, ax = plt.subplots(1)
ax.plot(x1, x2)
ax.set_aspect(1)

plt.grid(linestyle='--')

plt.title('Nom et prénom 2GT18', fontsize=10)

plt.show()
```



Activité N°4 Avec cent points répartis autour du cercle, la figure se fond presque parfaitement dans la courbe idéale. Le polygone n'a plus vraiment d'arêtes visibles : l'approximation discrète se dissout dans une rondeur presque continue. C'est un petit rappel que le cercle n'est finalement qu'une limite, un mirage approché par une multitude de segments minuscules. L'ordinateur, en bon artisan patient, tisse ici une frontière douce entre géométrie exacte et représentation numérique, une zone fertile pour expérimenter encore d'autres formes.

```
import numpy as np
import matplotlib.pyplot as plt

plt.axis([-1.25,1.25,-1.25,1.25])

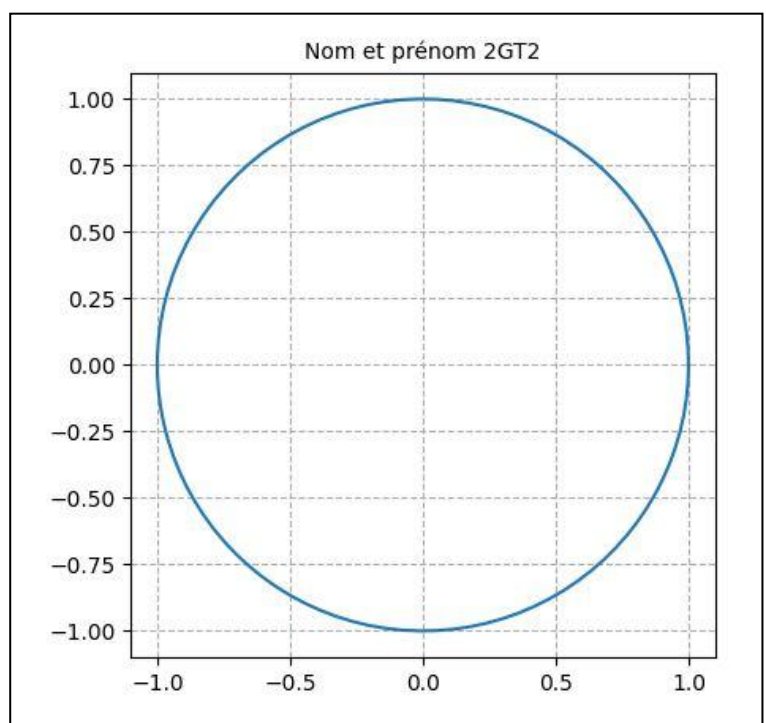
a = np.linspace(0, 2*np.pi, 100)
r = np.sqrt(1.0)
x1 = r*np.cos(a)
x2 = r*np.sin(a)

fig, ax = plt.subplots(1)
ax.plot(x1, x2)
ax.set_aspect(1)

plt.grid(linestyle='--')

plt.title('Nom et prénom 2GT18', fontsize=10)

plt.show()
```



Un algorithme qui affiche les quatre formes correspondant aux quatre activités, disposées verticalement les unes sous les autres :

```
import numpy as np
import matplotlib.pyplot as plt

# Une seule figure avec deux zones (2 lignes, 1 colonne)
fig, (ax1, ax2, ax3, ax4) = plt.subplots(4, 1)

# --- ACTIVITE N°1 ---
ax1.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 5)
r = np.sqrt(1.0)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
ax1.plot(x1, x2)
ax1.set_aspect(1)
ax1.grid(linestyle='--')
ax1.set_title('Nom et prénom 2GT18', fontsize=10)

# --- ACTIVITE N°2 ---
ax2.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 7)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
ax2.plot(x1, x2)
ax2.set_aspect(1)
ax2.grid(linestyle='--')
ax2.set_title('Nom et prénom 2GT18', fontsize=10)

# --- ACTIVITE N°3 ---
plt.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 10)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
ax3.plot(x1, x2)
ax3.set_aspect(1)
plt.grid(linestyle='--')
ax3.set_title('Nom et prénom 2GT18', fontsize=10)

# --- ACTIVITE N°4 ---
plt.axis([-1.25,1.25,-1.25,1.25])
a = np.linspace(0, 2*np.pi, 100)
x1 = r*np.cos(a)
x2 = r*np.sin(a)
ax4.plot(x1, x2)
ax4.set_aspect(1)
plt.grid(linestyle='--')
ax4.set_title('Nom et prénom 2GT18', fontsize=10)

plt.tight_layout()
plt.show()
```