ПРАКТИЧНА РОБОТА 10
Побудова простих графічних інтерфейсів та візуалізація даних
Система оцінювання

№	Тема	К-ть балів
1.	*Захист принаймні одного завдання з роботи*	1
2.	Практичні завдання	3,6*
3.	*Здача звіту*	0,4
	Всього	**5**

* – діє бонусна система

1. ***0,5 бала*** Зберіть текстовий редактор – програмний проект з глави 2 книги «Tkinter GUI Application Development Blueprints» за авторством Б. Чаударі (B. Chaudhary). Детально опишіть у звіті основні будівельні блоки програмного коду.
2. ***0,5 бала*** Зберіть цифрову барабанну установку (drum machine) – програмний проект з глави 3 книги «Tkinter GUI Application Development Blueprints» за авторством Б. Чаударі (B. Chaudhary). Детально опишіть у звіті основні будівельні блоки програмного коду.
3. ***0,5 бала*** Зберіть додаток для гри в шахи – програмний проект з глави 4 книги «Tkinter GUI Application Development Blueprints» за авторством Б. Чаударі (B. Chaudhary). Детально опишіть у звіті основні будівельні блоки програмного коду.
4. ***0,5 бала*** Зберіть аудіоплеєр – програмний проект з глави 5 книги «Tkinter GUI Application Development Blueprints» за авторством Б. Чаударі (B. Chaudhary). Детально опишіть у звіті основні будівельні блоки програмного коду.
5. ***0,5 бала*** Зберіть графічний редактор – програмний проект з глави 6 книги «Tkinter GUI Application Development Blueprints» за авторством Б. Чаударі (B. Chaudhary). Детально опишіть у звіті основні будівельні блоки програмного коду.
6. ***0,3 бала*** За допомогою функції plot() побудуйте графіки відповідно до свого варіанту

№	Функція	№	Функція
1.	$f(x)=\frac{1}{10}\left(x-\frac{x^{\frac{3}{2}}}{10}\right)^2+\arcsin(2x+1)$	7.	$f(x)=2.87^{x^3}-3.06x^2+\cos\left(\frac{x^2+1}{3}-\log_2(\sin x^4)\right)$
2.	$f(x)=-1.7\pi x^3+2\arccos 7e^{-x}+3\ln\left\lvert 5\mathrm{tg}\, x^{\frac{2}{3}}\right\rvert$	8.	$f(x)=\log_2(7x\cos(arcctg\,(2x^5-1)+7x)$
3.	$f(x)=\sin x\cos x+x^5-12\ln 3x$	9.	$f(x)=\sec(\lg 3x-\cos 2x)^3-\sqrt[4]{x^3-2x+7}$
4.	$f(x)=\sin x^{\ln 5-\sqrt{2x^3}}+e^{x-1}$	10.	$f(x)=4x^3-\sqrt{\ln(13x-sh\,x)+arctg(x+1)}$
5.	$f(x)=7x^2\sin\left(\frac{1}{7x^2}\right)+sh\,5x^3$	11.	$f(x)=-2.5\sin x^3+e^{-\sqrt{ctg\,5x^2}}+1$
6.	$f(x)=x-\frac{\cos x-\sin x}{\sin x+\cos x}+e^{-3x^4+2}$	12.	$f(x)=tg(x^2-5x+7)+e^{\sin(\arccos(5x)-1)}$

Бібліотека matplotlib для Windows зазвичай інсталюється в таких програмних комплексах, як Anaconda, Enthought Canopy, Algorete Loopy та ін. Приклад роботи з середовищем розробки Spyder з Anaconda наведено на рис. 1

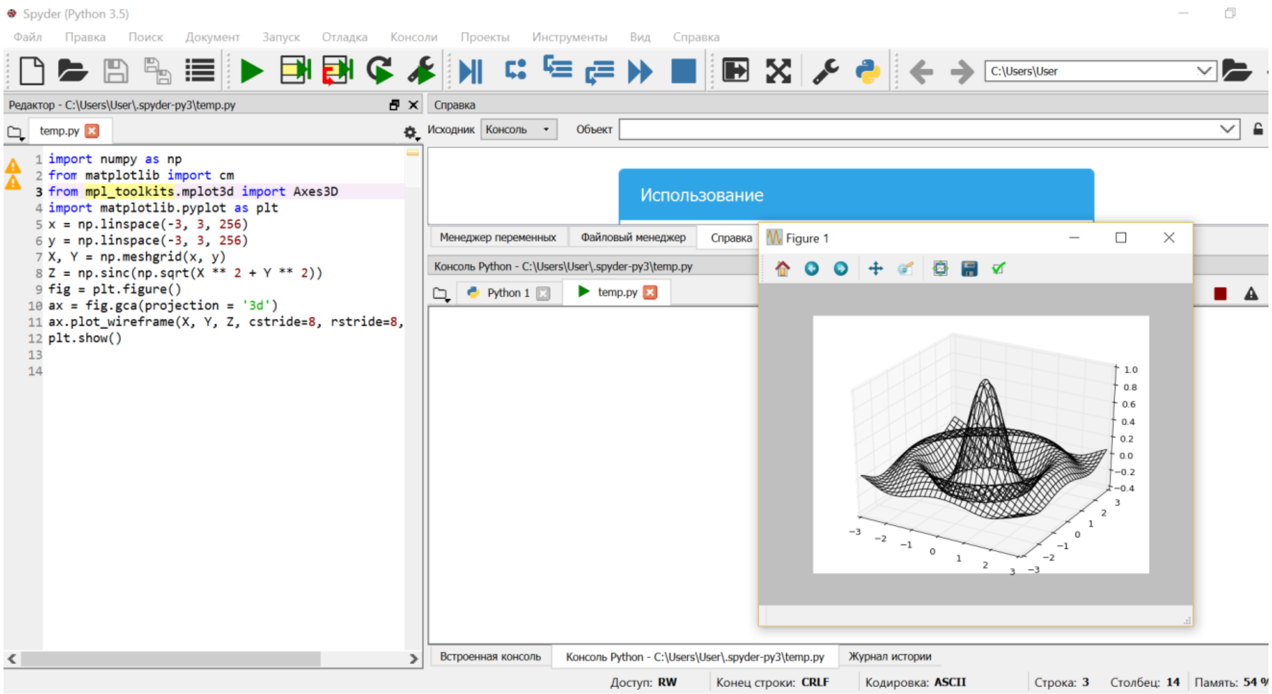


Рис. 1. Робота у Spider

Для підключення модулю для побудови графіків у код скрипту на початку дописують

```
import matplotlib.pyplot as plt
```

Дуже поширеною практикою є заміна назви matplotlib.pyplot на еквівалентну – plt, щоб скоротити код. Побудова графіків для функцій виду $y = f(x)$ реалізується за допомогою методу plot(), який приймає 2 параметри: набір значень x та набір значень y:

```
X = range(100)
Y = [value ** 2 for value in X]
plt.plot(X, Y)
plt.show()
```

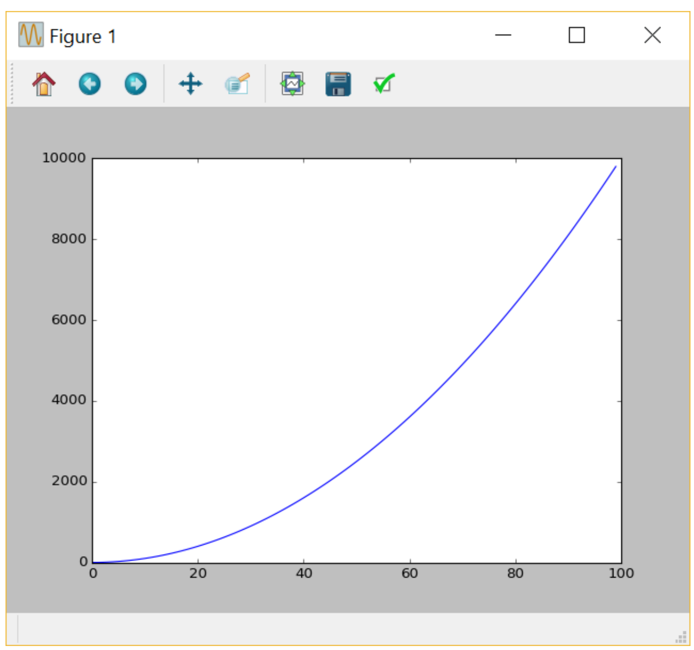


Рис. 2. Отриманий графік $y = f(x)$

7. *0,4 бала* Побудуйте графік кривої відповідно до свого варіанту.

№	Назва	Функція	Система координат	Проміжок
1.	Серце	$(x^2 + y^2 - 1)^3 = x^2 y^3$	Декартова	Підібрати
2.	Квітка	$r = \sqrt{1 + \cos(6t)}$	Полярна	$t \in [-4\pi, 4\pi]$
3.	Каннабола	$r = (1 + 0.9\cos(8t)) \cdot (1 + 0.1\cos(24t)) \cdot (1 + \sin(t)) \cdot (1 - 0.02\sin(200t))$	Полярна	$t \in [0, 2\pi]$
4.	Овал Кассіні	$(x^2 + y^2)^2 - 2c^2(x^2 - y^2) = a^4 - c^4$	Декартова	Підібрати
5.	Овал Кассіні	$\rho^4 - 2c^2\rho^2\cos(2\varphi) = a^4 - c^4$	Полярна	Підібрати
6.	Фігури Ліссажу	$x(t) = A\sin(at + \delta)$ $y(t) = B\sin(bt)$	Декартова	Підібрати
7.	Логарифмічна крива	$r = ae^{b\theta}$ або $\theta = \frac{1}{b}\ln(r/a)$	Полярна	Підібрати
8.	Логарифмічна крива	$x(t) = r\cos t = ae^{bt}\cos t,$ $y(t) = r\sin t = ae^{bt}\sin t$	Декартова	Підібрати
9.	Троянда	$\rho = a\sin(k\varphi)$	Полярна	Підібрати
10.	Равлик Паскаля	$(x^2 + y^2 - ay)^2 = l^2(x^2 + y^2)$	Декартова	Підібрати
11.	Равлик Паскаля	$\rho = l - a\sin\varphi$	Полярна	Підібрати
12.	Синусоїдальна спіраль	$r^n = a^n\cos(n\varphi)$	Полярна	Підібрати

Полярна система координат використовується для побудови графіків, що відображають процеси та явища, що залежать від кутів. Наприклад, потужність динаміка, яка залежить від кута, з якого її вимірюють. Також циклічні дані, на зразок щорічної чи щоденної статистики, можна зручно зобразити в полярних координатах.

Побудуємо графік функції, записаної в полярних координатах (рис. 3)

```
import numpy as np
import matplotlib.pyplot as plt
T = np.linspace(0 , 2 * np.pi, 1024)
plt.axes(polar = True)
plt.plot(T, 1. + .25 * np.sin(16 * T), c= 'k')
plt.show()
```

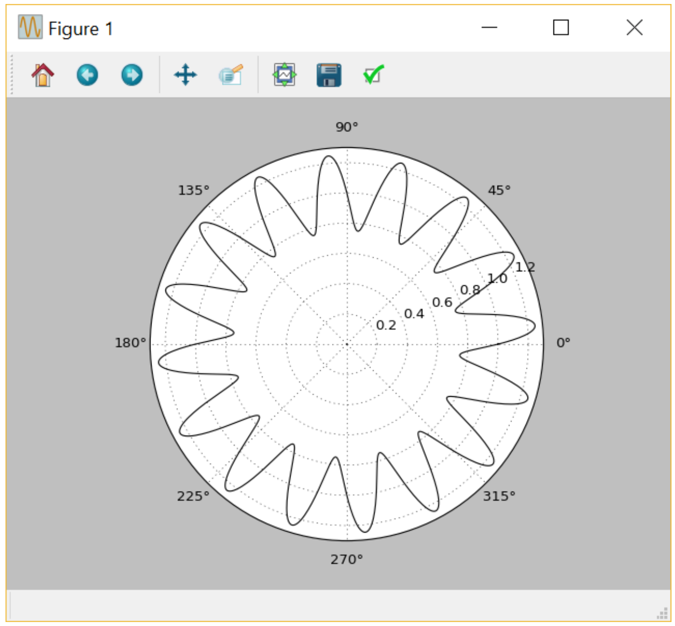


Рис. 3. Графік функції $\rho(\varphi) = 1.0 + 0.25\sin 16\varphi$

Екземпляр Axes явно створюється за допомогою pyplot.axes(). За допомогою опційного параметра polar буде задано виконання проекції в полярні координати.

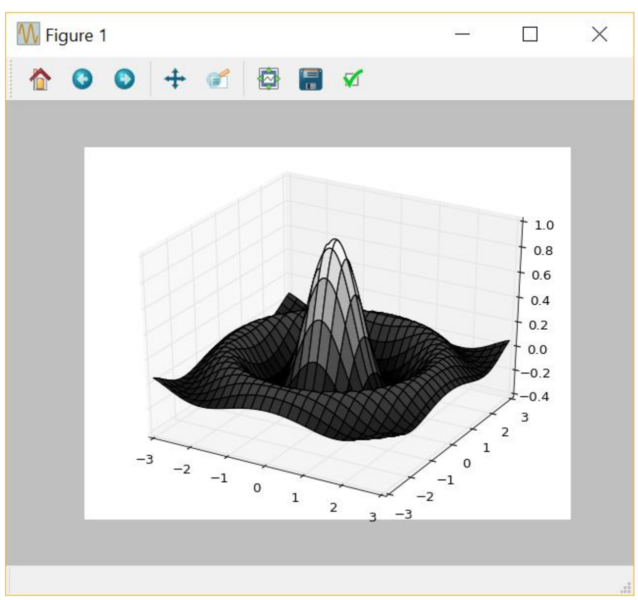
8. ***0,4 бала*** Побудуйте поверхню відповідно до свого варіанту. Відобразіть значення, що використовувалися при побудові.

№	Назва	Функція	Система координат	Проміжок
1.	Поверхня Діні	$x(u,v) = \cos(u)\sin(v)$ $y(u,v) = \sin(u)\sin(v)$ $z(u,v) = \cos(v) + \ln\left(\tan\left(\frac{v}{2}\right)\right) + 0.2u$	Декартова	$u \in [0, 4\pi]$ $v \in [1e-3, 2]$
2.	Мушля	$\begin{cases} x = \frac{5}{4}\left(1-\frac{v}{2\pi}\right)\cos 2v\,(1+\cos u) + \cos 2v \\ y = \frac{5}{4}\left(1-\frac{v}{2\pi}\right)\sin 2v\,(1+\cos u) + \sin 2v \\ z = \frac{10v}{2\pi} + \frac{5}{4}\left(1-\frac{v}{2\pi}\right)\sin u + 15 \end{cases}$	Декартова	$0 \le u \le 2\pi,$ $-2\pi \le v \le 2\pi$
3.	Мавпяче сідло	$z = x^3 - 3xy^2$	Декартова	Підібрати
4.	Катеноїд	$\begin{cases} x = \mathrm{ch}\,(u)\cos v \\ y = \mathrm{ch}\,(u)\sin v \\ z = u \end{cases}$	Декартова	$v \in [0, 2\pi)$ $u \in R$
5.	Гіперболічний параболоїд	$z = \frac{x^2}{a^2} - \frac{y^2}{b^2}$	Декартова	Підібрати
6.	Парасолька Уітні	$\begin{cases} x = uv \\ y = u \\ z = v^2 \end{cases}$	Декартова	Підібрати
7.	Стрічка Мебіуса	$\begin{cases} x = \left(1 + \frac{v}{2}\cos\left(\frac{u}{2}\right)\right)\cos u \\ y = \left(1 + \frac{v}{2}\cos\left(\frac{u}{2}\right)\right)\sin u \\ z = \frac{v}{2}\sin\frac{u}{2} \end{cases}$	Декартова	Підібрати

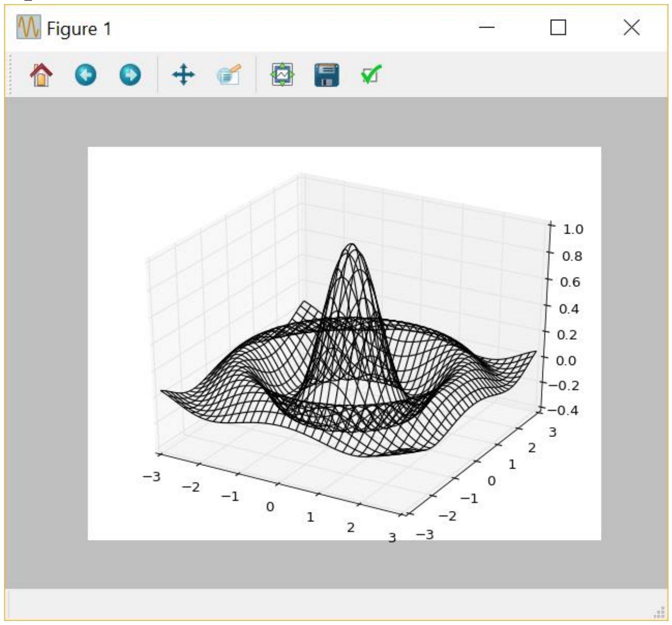
8.	Тор	$\begin{cases} x(\varphi,\psi) = (R + r\cos\varphi)\cos\psi \\ y(\varphi,\psi) = (R + r\cos\varphi)\sin\psi \\ z(\varphi,\psi) = r\sin\varphi \end{cases}$	Декартова	$\varphi, \psi \in [0, 2\pi)$
9.	Ластів'ячий хвіст	$\begin{cases} x = u \\ y = 2v^3 + uv \\ z = 3v^4 + uv^2 \end{cases}$	Декартова	Підібрати
10.	Коноїд	$\begin{cases} x = v\cos u + lf(u) \\ y = v\sin u + mf(u), \\ z = nf(u) \end{cases}$ $f(u)$ – деяка функція	Декартова	Підібрати
11	Пляшка Клейна	$\begin{cases} x = \left(r + \cos\frac{u}{2}\sin v - \sin\frac{u}{2}\sin 2v\right)\cos u \\ y = \left(r + \cos\frac{u}{2}\sin v - \sin\frac{u}{2}\sin 2v\right)\sin u \\ z = \sin\frac{u}{2}\sin v + \cos\frac{u}{2}\sin 2v \end{cases}$	Декартова	Підібрати
12.	Гелікоїд	$\begin{cases} x = u\cos v \\ y = u\sin v \\ z = hv \end{cases}$	Декартова	Підібрати

Як і раніше, згенеруємо певні тестові дані, встановимо екземпляр (instance) Axes3D та передамо йому дані:

```
import numpy as np
from matplotlib import cm
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
x = np.linspace(-3, 3, 256)
y = np.linspace(-3, 3, 256)
X, Y = np.meshgrid(x, y)
Z = np.sinc(np.sqrt(X ** 2 + Y ** 2))
fig = plt.figure()
ax = fig.gca(projection = '3d')
ax.plot_surface(X, Y, Z, cmap=cm.gray)
plt.show()
```



Дві матриці, X та Y, створені для зберігання координат для регулярної сітки. Обчислюємо матрицю Z та скалярне поле функції від X та Y. Викликаємо метод plot_surface(), який приймає X, Y, Z, щоб відобразити скалярне поле як 3D-поверхню. Кольори обираються з colormap (опційний параметр cmap) відповідно до значень з матриці Z.



Побудова параметричної тривимірної поверхні

До цього функція plot_surface() використовувалась для побудови графіку для скалярного поля (функції у вигляді f(x, y) = z). Проте бібліотека matplotlib дозволяє будувати загальні, параметричні тривимірні поверхні. Продемонструємо це, нарисувавши тор:

```
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
# Генерація сітки (mesh) для тору
angle = np.linspace(0, 2 * np.pi, 32)
theta, phi = np.meshgrid(angle, angle)
r, R = .25, 1.
X = (R + r * np.cos(phi)) * np.cos(theta)
Y = (R + r * np.cos(phi)) * np.sin(theta)
Z = r * np.sin(phi)
# Показ сітки (mesh)
fig = plt.figure()
ax = fig.gca(projection = '3d')
ax.set_xlim3d(-1, 1)
ax.set_ylim3d(-1, 1)
ax.set_zlim3d(-1, 1)
ax.plot_surface(X, Y, Z, color = 'w', rstride = 1, cstride =
1)
plt.show()
```

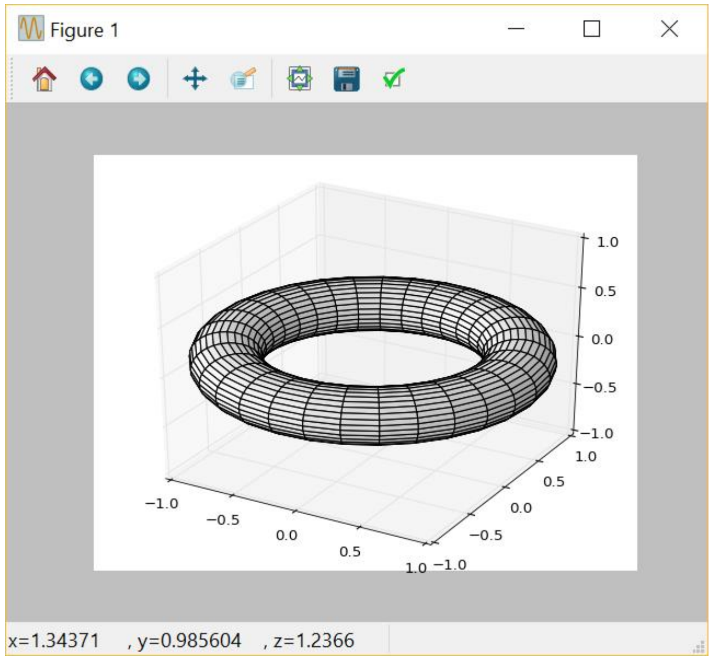


Figure 1
1.0
0.5
0.0
−0.5
−1.0
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
x=1.34371 , y=0.985604 , z=1.2366