VPython - symulacje fizyczne z grafiką 3D dla każdego

wykład 8

Dr hab. Adam Bzdak, prof. AGH

Legenda

legend_details.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\legend_details.py (2.7.12)

File Edit Format Run Options Window Help

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

plt.rcParams['font.size'] = 18
plt.rcParams['legend.fontsize'] = 18

x = np.linspace(0,10,50)
y1 = np.sin(x); y2 = np.cos(x); y3 = np.exp(-x)

plt.plot(x, y1, 'k-', label='sin(x)', lw=2)
plt.plot(x, y2, 'b', dashes=[20,10], label='cos(x)', lw=2)
plt.plot(x[::2], y3[::2], 'rs-', label='exp(-x)', ms=8, lw=2)

plt.legend(loc = 'lower right',
           ncol=2, numpoints=1, labelspacing=0.5,
           handletextpad=0.5, handlelength=2.5,
           borderaxespad=0.5, borderpad=0.05,
           columnspacing=0.9, frameon=False)

plt.axis([-0.5, 10.5, -1.7, 1.5])
plt.show()
```

plt.style.use('classic')

co drugi element

proszę pobawić się liczbami

Ln: 11 Col: 27

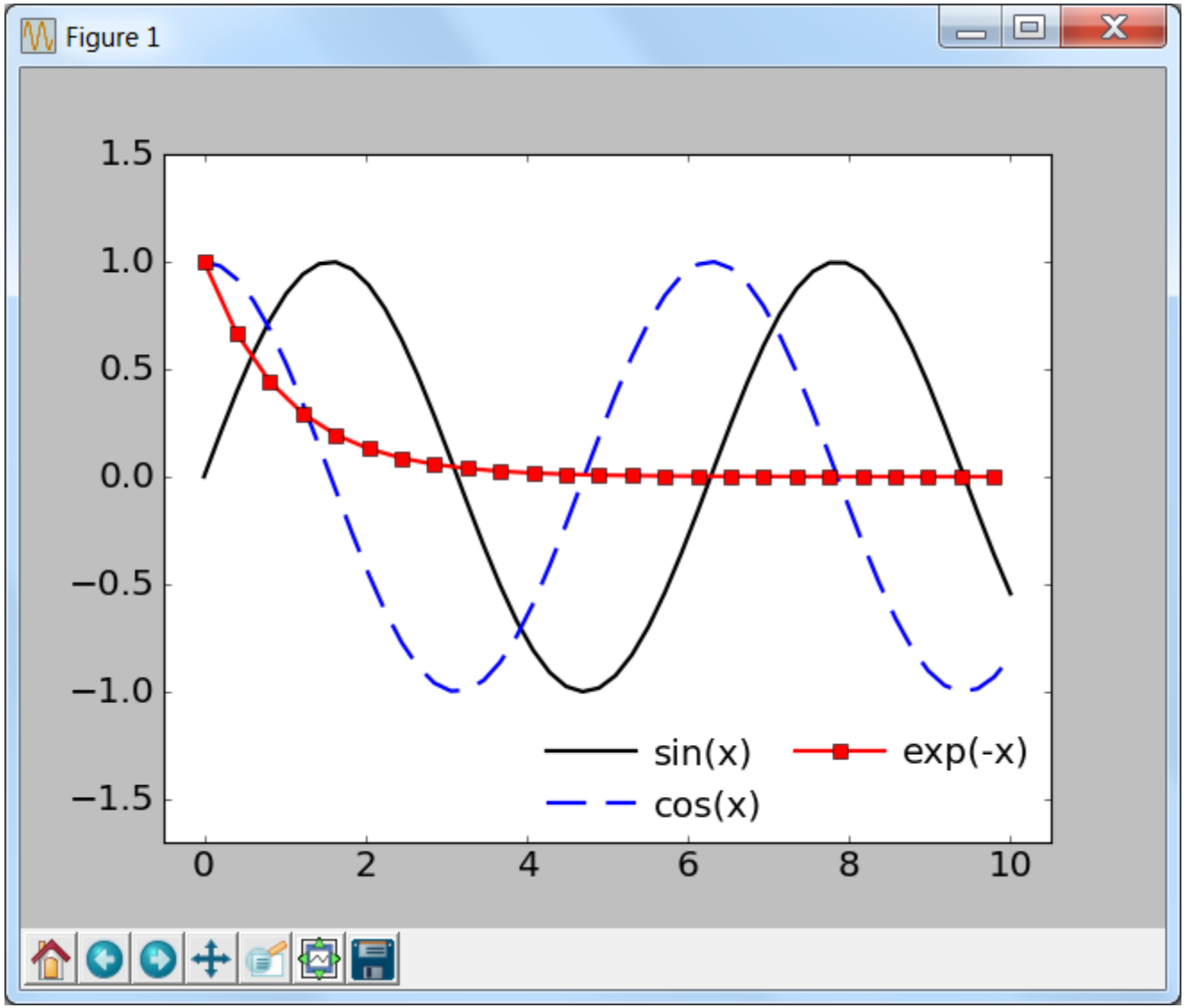


Figure 1
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
2
4
6
8
10
sin(x)
cos(x)
exp(-x)

Ustawienia

general_settings.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\general_settings.py (2.7.12)

File Edit Format Run Options Window Help

```
import matplotlib.pyplot as plt

# print plt.rcParams.keys()

# plt.rcParams['font.family'] = 'Arial'
# plt.rcParams['font.family'] = 'Serif'
plt.rcParams['font.family'] = 'Times New Roman'
plt.rcParams['font.size'] = 28
plt.rcParams['legend.fontsize'] = 26
plt.rcParams['lines.linewidth'] = 2.5
plt.rcParams['axes.linewidth'] = 1.0
plt.rcParams['xtick.major.size'] = 8.0
plt.rcParams['xtick.major.width'] = 1.0
plt.rcParams['ytick.major.size'] = 8.0
plt.rcParams['ytick.major.width'] = 1.0
plt.rcParams['xtick.minor.size'] = 4.0
plt.rcParams['xtick.minor.width'] = 1.0
plt.rcParams['ytick.minor.size'] = 4.0
plt.rcParams['ytick.minor.width'] = 1.0
# plt.rcParams['xtick.direction'] = 'out'
# plt.rcParams['ytick.direction'] = 'out'
```

Ln: 23 Col: 0

Wykres 3D

3d_1.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\3d_1.py (2.7.12)

File Edit Format Run Options Window Help

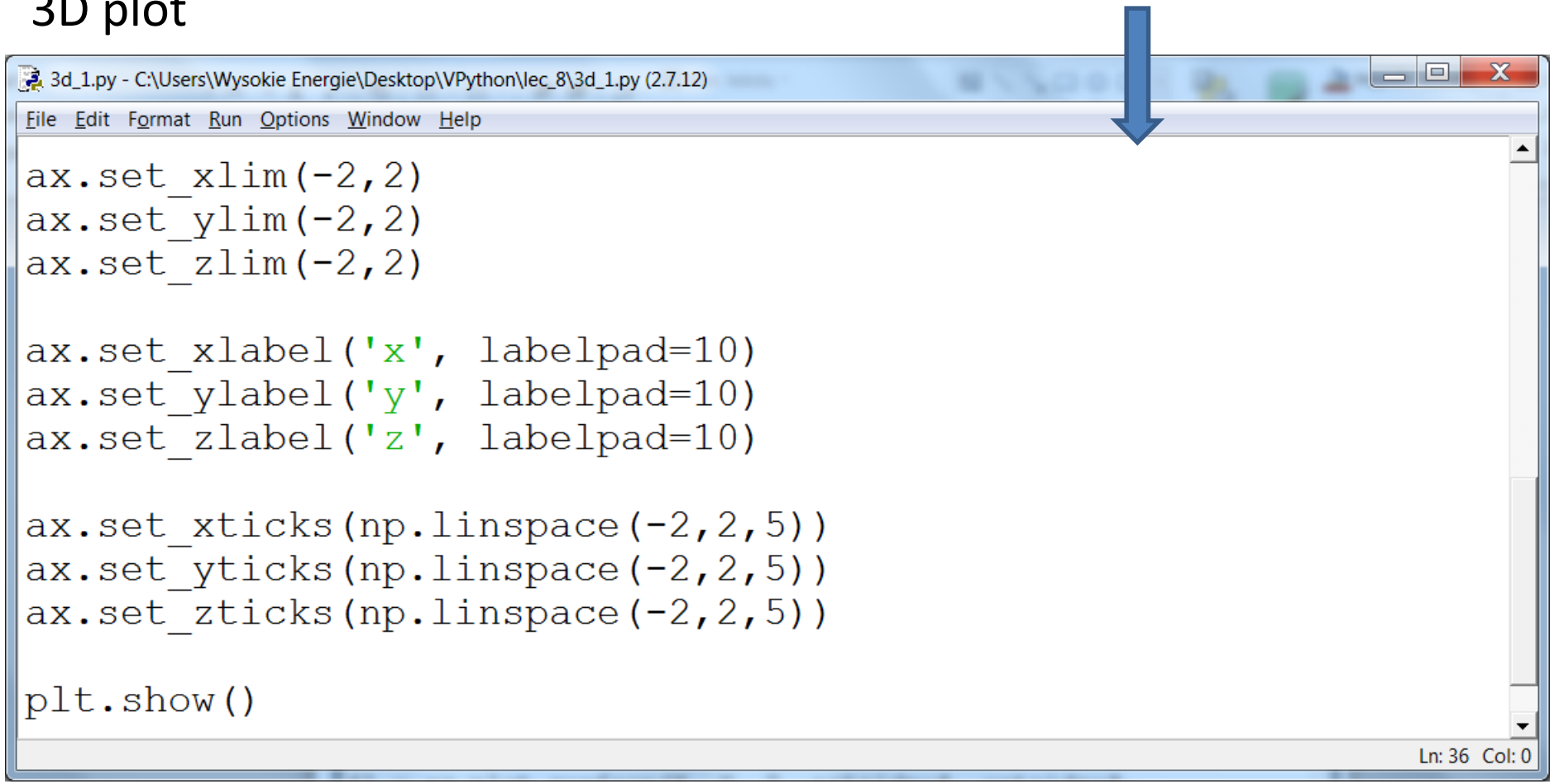
```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

plt.rcParams['font.size'] = 18

fig = plt.figure(figsize=(11, 8))
ax = fig.add_subplot(111, projection='3d')

x = np.linspace(-2,2,100)
y = np.linspace(-2,2,100)

(X,Y) = np.meshgrid(x,y)
Z = np.cos(X**2 + Y**2)

f1 = ax.plot_surface(X, Y, Z, rstride=4, cstride=4,
                     linewidth=0.1, cmap='rainbow')

fig.colorbar(f1, shrink=0.5, aspect=15,
             ticks=np.linspace(-1,1,6))
```

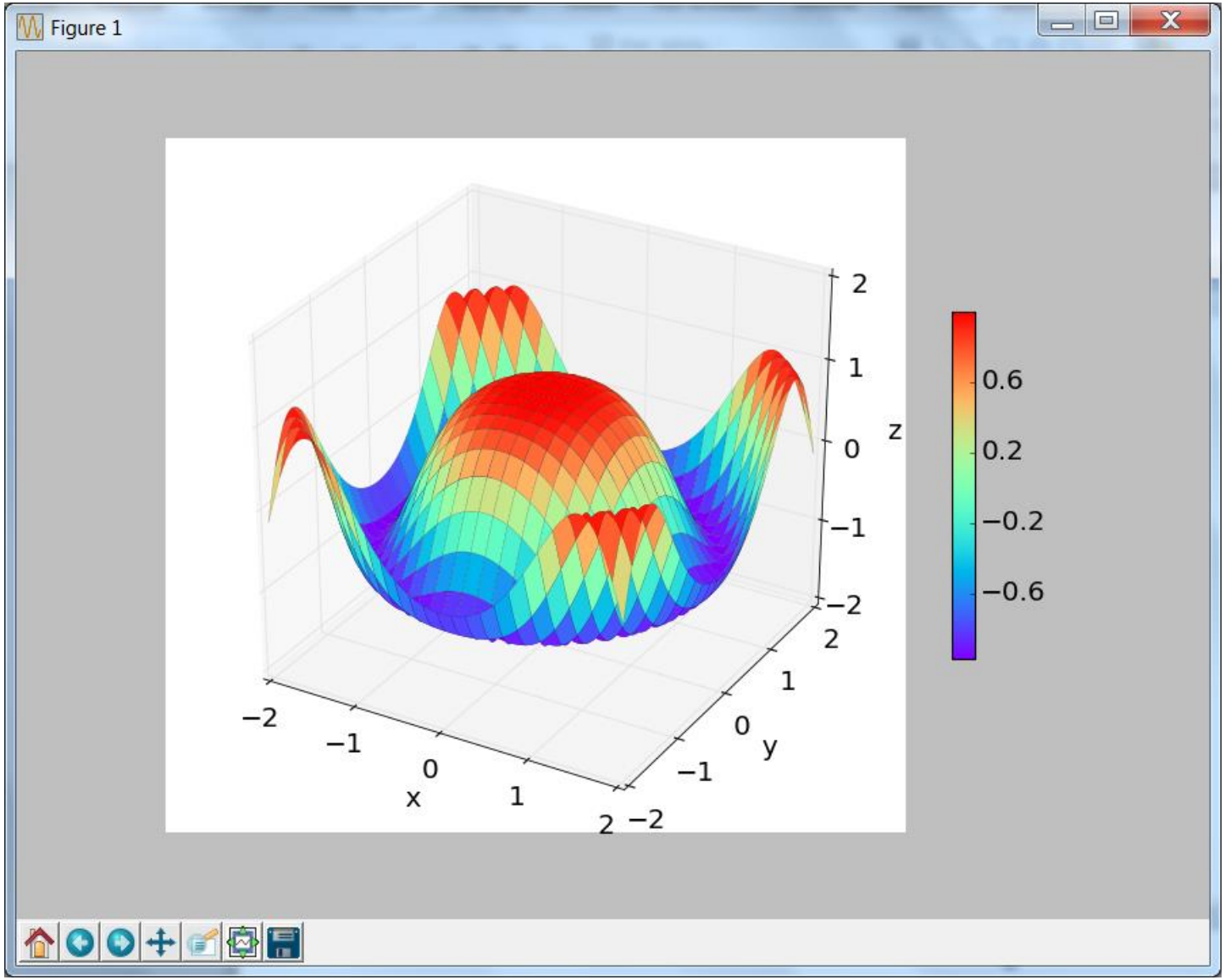
1 rząd, 1 kolumna, wykres 1

100/4 linii

Ln: 17 Col: 0

3D plot


3d_1.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\3d_1.py (2.7.12)
File Edit Format Run Options Window Help
ax.set_xlim(-2,2)
ax.set_ylim(-2,2)
ax.set_zlim(-2,2)
ax.set_xlabel('x', labelpad=10)
ax.set_ylabel('y', labelpad=10)
ax.set_zlabel('z', labelpad=10)
ax.set_xticks(np.linspace(-2,2,5))
ax.set_yticks(np.linspace(-2,2,5))
ax.set_zticks(np.linspace(-2,2,5))
plt.show()
Ln: 36 Col: 0



Za pomocą myszki można kręcić, powiększać itp.

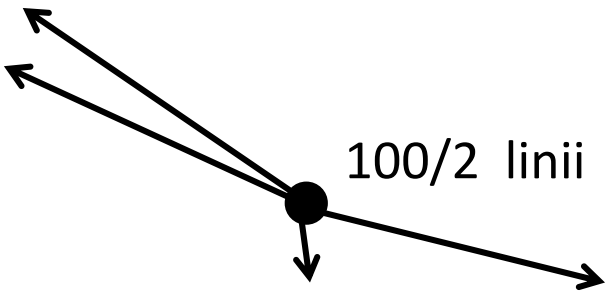
wykres 3D, wireframe

3d_2.py - C:/Users/Wysokie Energie/Desktop/VPython/lec_8/3d_2.py (2.7.12)

File Edit Format Run Options Window Help

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

plt.rcParams['font.size'] = 18

fig = plt.figure(figsize=(11, 8))
ax = fig.add_subplot(111, projection='3d')

x = np.linspace(-2,2,100)
y = np.linspace(-2,2,100)

(X,Y) = np.meshgrid(x,y)
Z = np.cos(X**2 + Y**2)

ax.plot_wireframe(X, Y, Z, rstride=2, cstride=2,
                  linewidth=1, color='black')
```

Ln: 21 Col: 7



reszta jak na str. 6

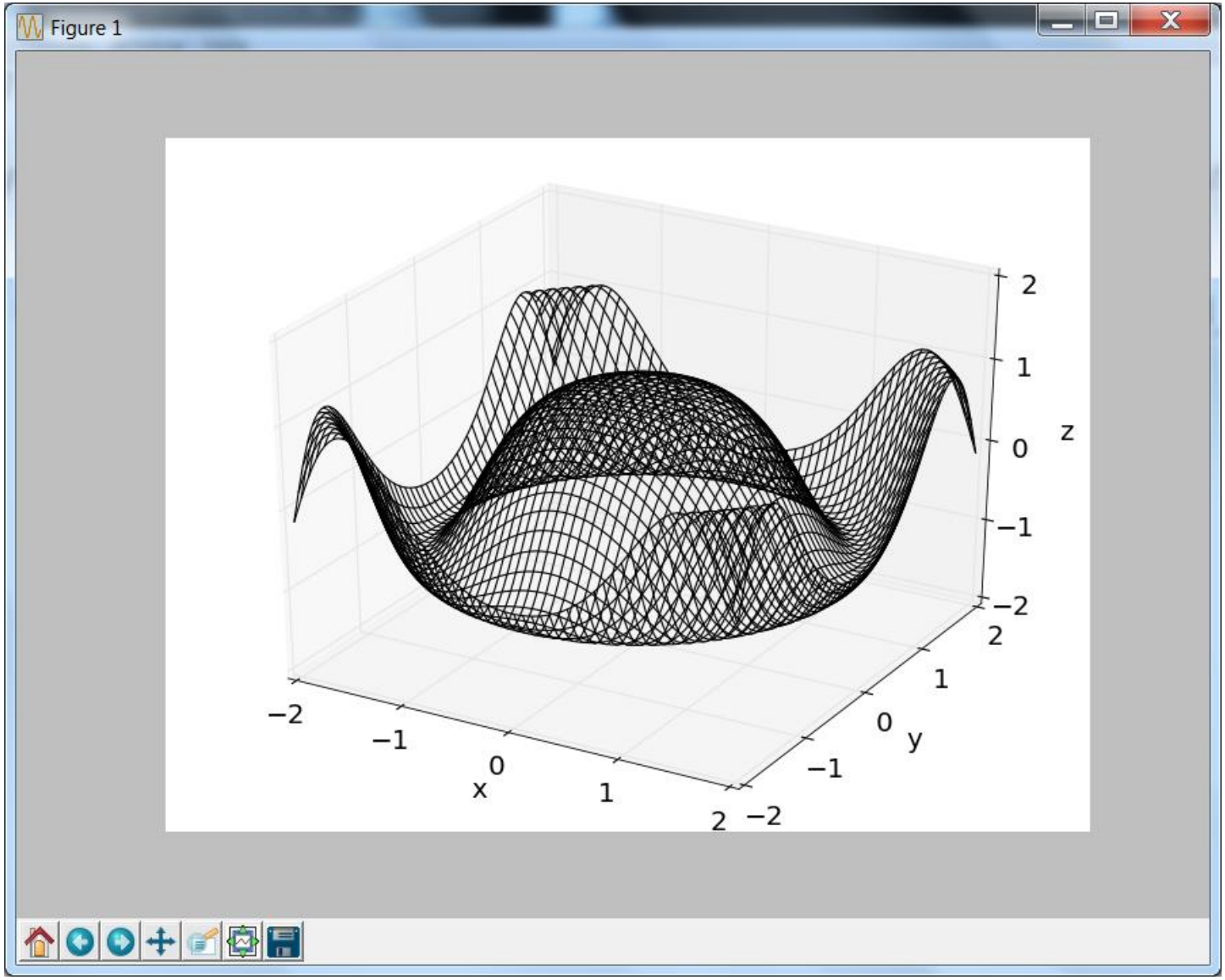


Figure 1
2
1
0
−1
−2
z
−2
−1
0
1
2
x
2
1
0
−1
−2
y

Siatka wykresów. GridSpec

grid_0.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\grid_0.py (2.7.12)

File Edit Format Run Options Window Help

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

sieć 2x1 (2 rzędy, 1 kolumna)

```
gs = gridspec.GridSpec(2, 1, height_ratios=[3, 1])


x = np.linspace(0,10,50)

ax1 = plt.subplot(gs[0,0])
ax1.plot(x, np.sin(x))

ax2 = plt.subplot(gs[1,0])
ax2.plot(x, np.cos(x))

plt.show()
```

rząd 0, kolumna 0

rząd 1, kolumna 1

Ln: 19 Col: 0

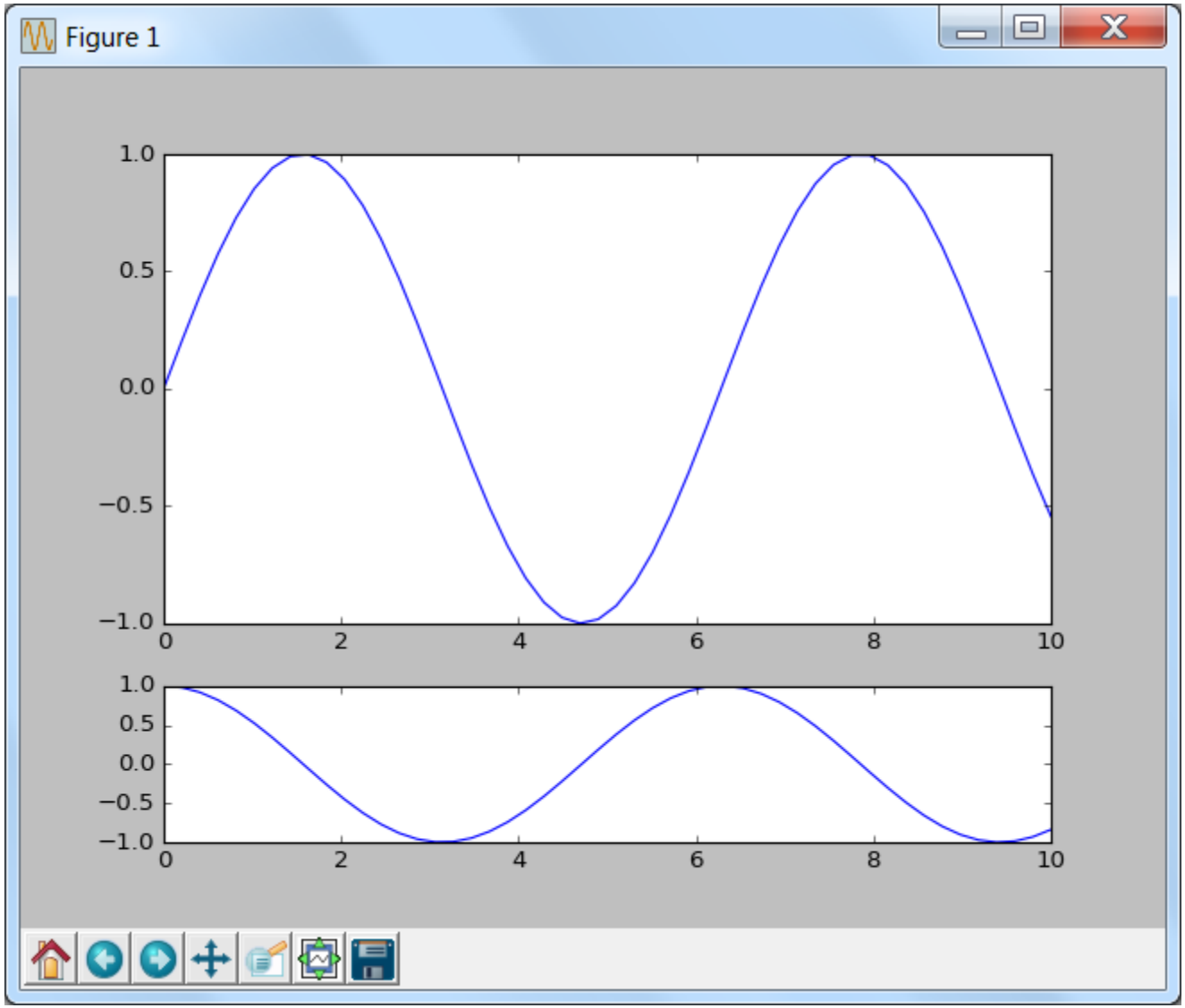


Figure 1
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10

GridSpec

grid_1.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\grid_1.py (2.7.12)

File Edit Format Run Options Window Help

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

gs = gridspec.GridSpec(2, 2)

x = np.linspace(0,10,50)

ax1 = plt.subplot(gs[0,:])
ax1.plot(x, np.sin(x))

ax2 = plt.subplot(gs[1,0])
ax2.plot(x, np.cos(x))

ax3 = plt.subplot(gs[1,1])
ax3.plot(x, np.exp(-x/2))

plt.show()
```

Ln: 16 Col: 23

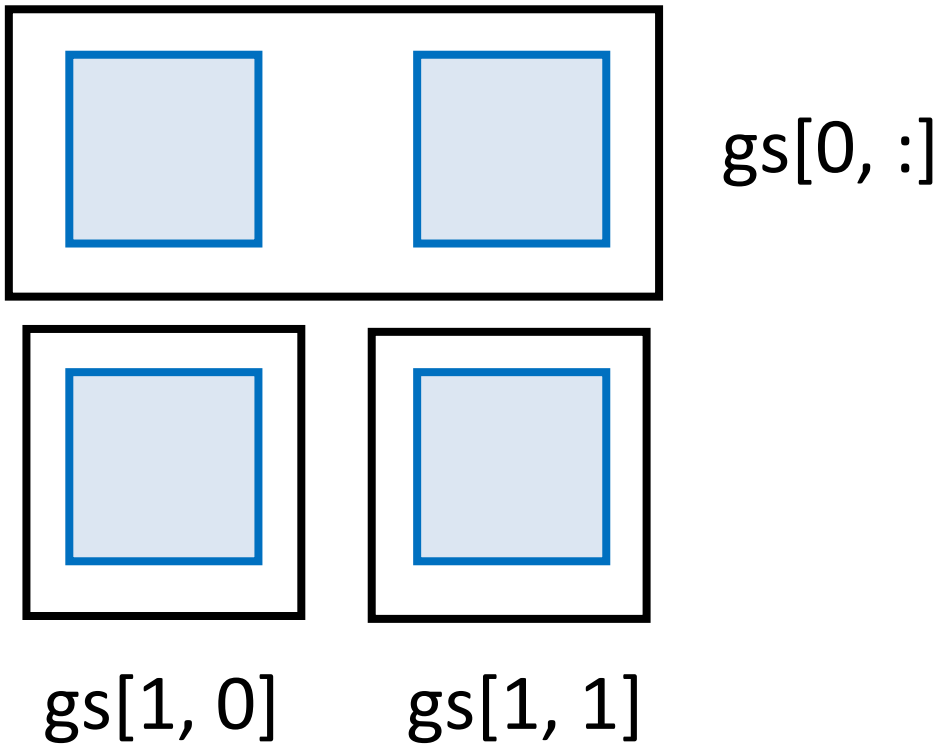
sieć 2x2 (2 rzędy, 2 kolumny)

rząd 0, wszystkie kolumny

rząd 1, kolumna 0

rząd 1, kolumna 1

sieć 2x2



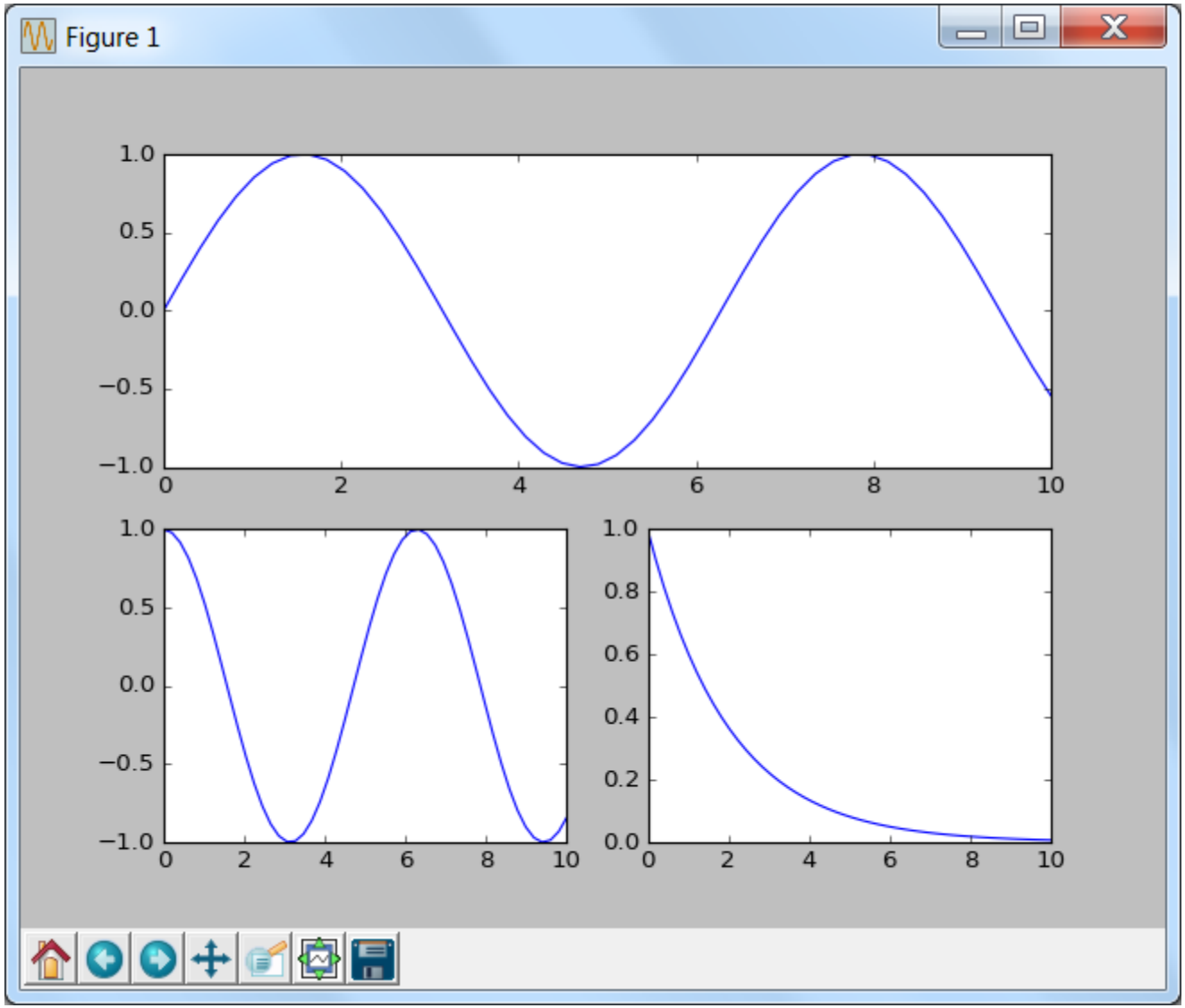


Figure 1
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
1.0
0.5
0.0
−0.5
−1.0
0
2
4
6
8
10
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10

GridSpec

grid_2.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\grid_2.py (2.7.12)

File Edit Format Run Options Window Help

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

gs = gridspec.GridSpec(3, 3, wspace=0.3, hspace=0.5,
                       height_ratios=[1, 1, 2])

x = np.linspace(0,10,10**2)

ax1 = plt.subplot(gs[0,:])
ax1.plot(x, np.sin(x))
ax1.semilogx()

ax2 = plt.subplot(gs[1,:-1])
ax2.plot(x, np.cos(x))

ax3 = plt.subplot(gs[1:,-1])
ax4 = plt.subplot(gs[2,0])
ax5 = plt.subplot(gs[2,1])

plt.show()
```

sieć 3x3

rząd 0; kolumny 0,1,2

rząd 1; kolumny 0, 1

rząd 1, 2; kolumna 2

Ln: 23 Col: 0

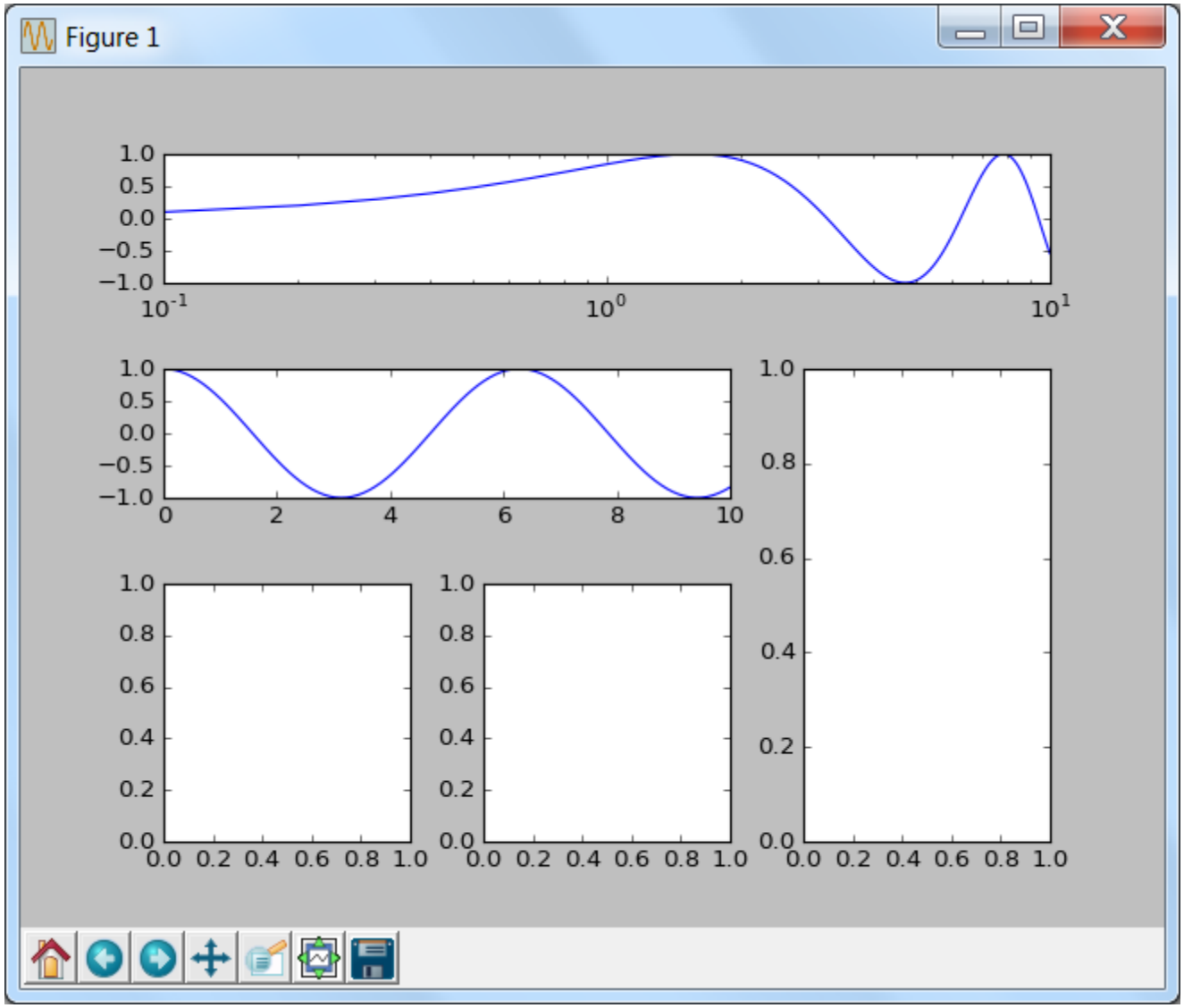
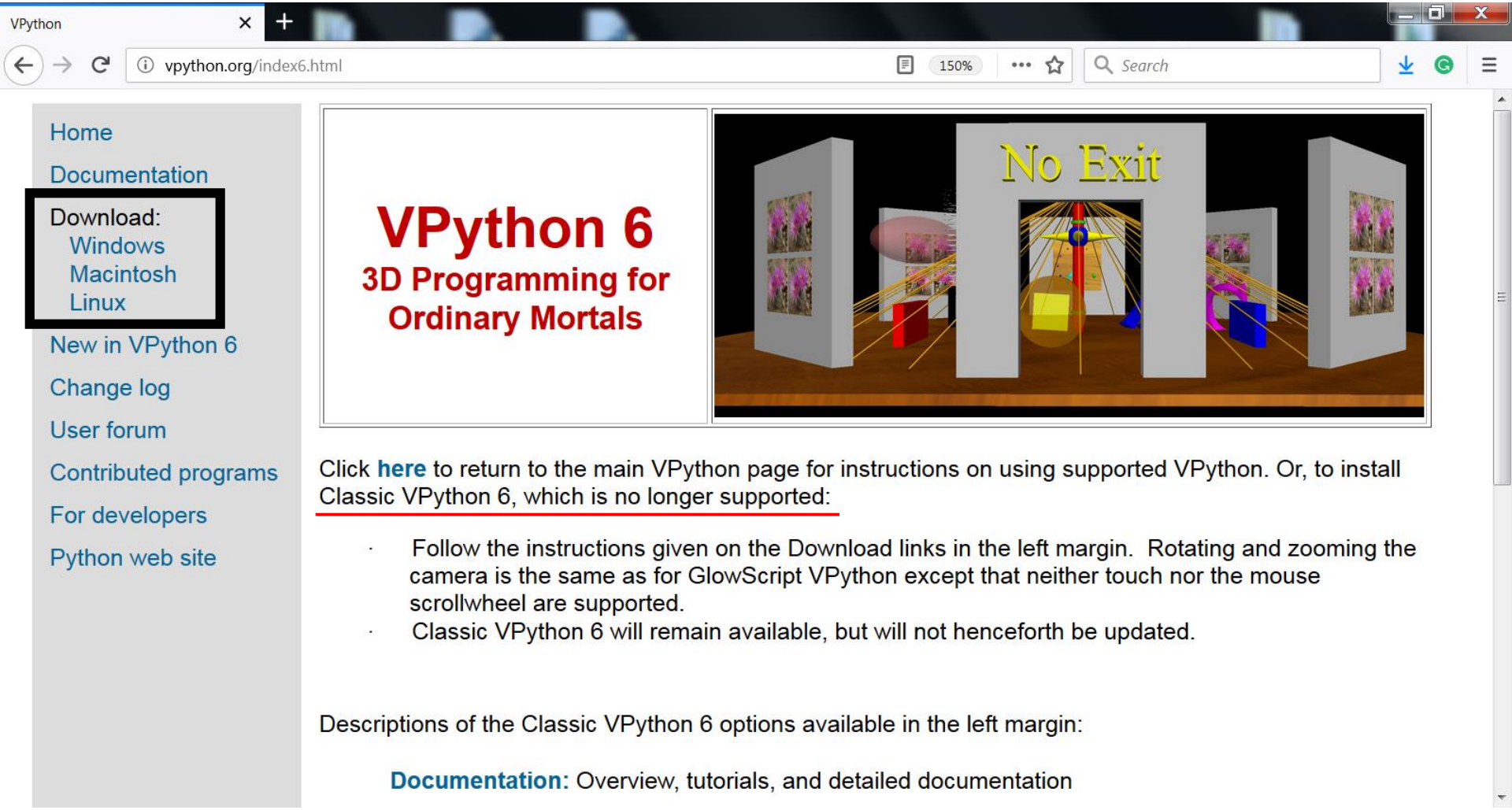


Figure 1
1.0
0.5
0.0
−0.5
−1.0
10^{-1}
10^{0}
10^{1}
0
2
4
6
8
10
0.8
0.6
0.4
0.2
0.0 0.2 0.4 0.6 0.8 1.0

VPython

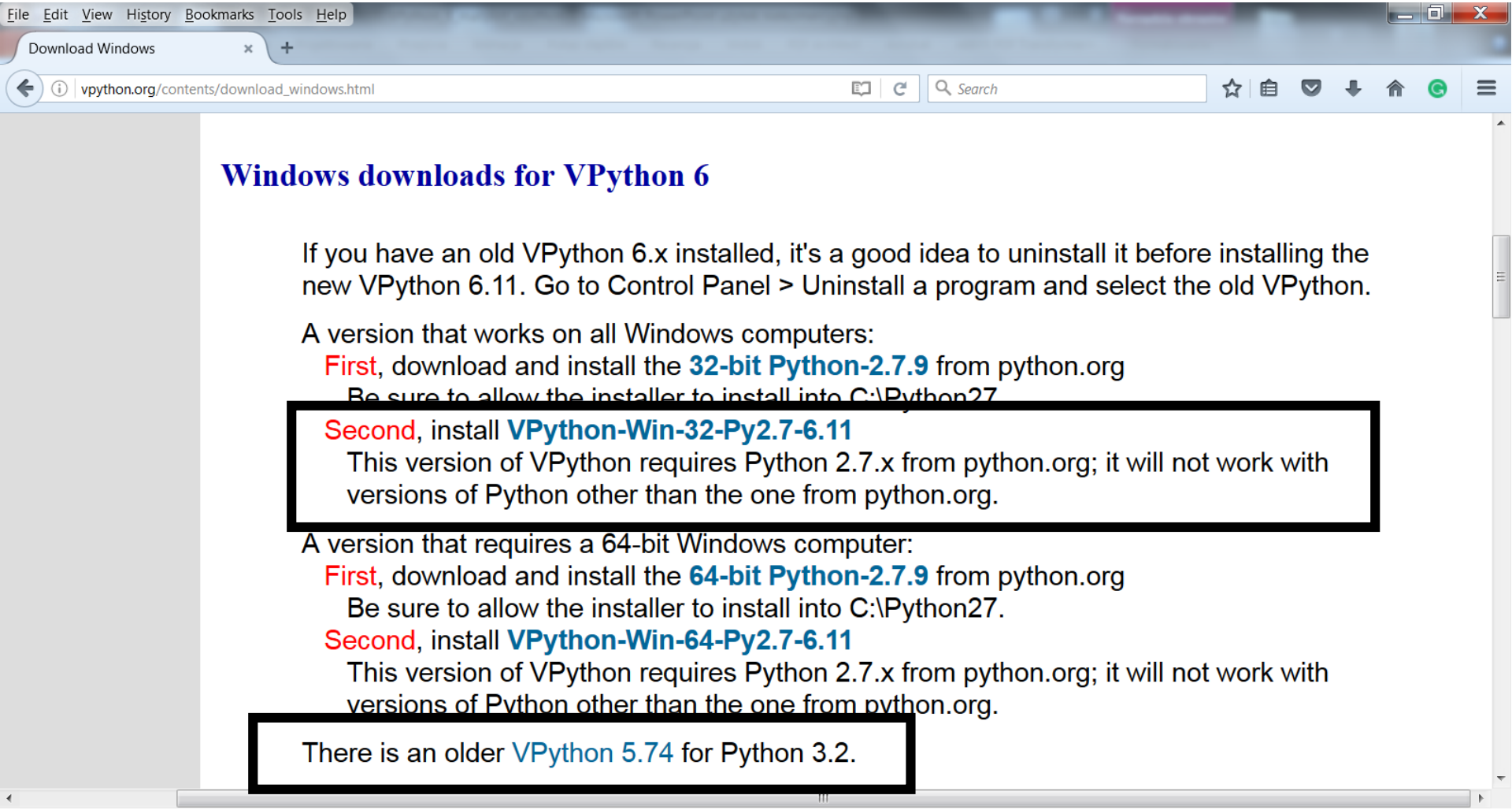
Classic VPython 6

http://vpython.org/index6.html



VPython 6 jest na wydziałowych komputerach. Lepiej używać VPython 7 ale dla naszych celów oba są dobre

Windows, dobre dla Python 2.7.x



Używamy VIDLE for VPython

Powinno pojawić się na pulpicie

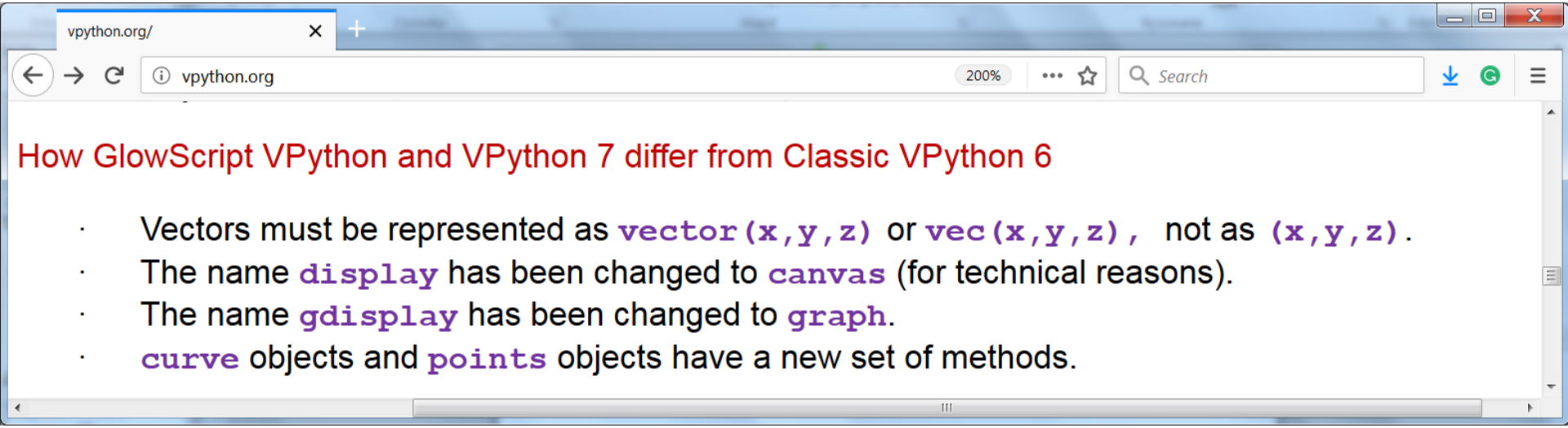
Mając Python 3.x lepiej zainstalować **VPython 7**

Robimy podobnie jak dla NumPy czy Matplotlib

python pip install vpython

Są inne opcje: proszę poczytać na http://vpython.org/

Drobne różnice pomiędzy classic a 7



Pierwszy program

vp_first_prog_1.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_1.py

File Edit Format Run Options Windows Help

```
from visual import *

scene = display()
scene.width = 550
scene.height = 580
scene.range = 2

ball = sphere()
```

scena

szerokość w pikselach

wysokość w pikselach

zasięg sceny (w stosunku do obiektów)

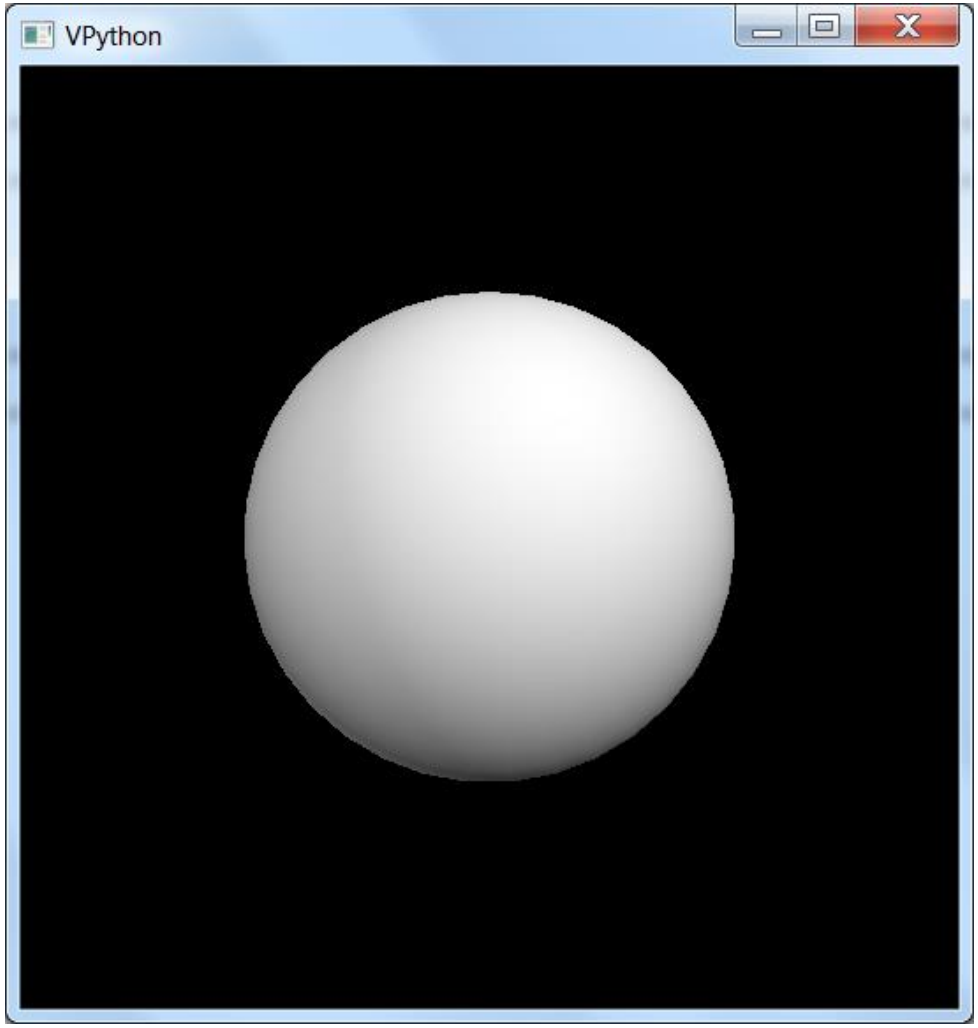
kula 3D, promień = 1

Ln: 11 Col: 0

VPython 7:

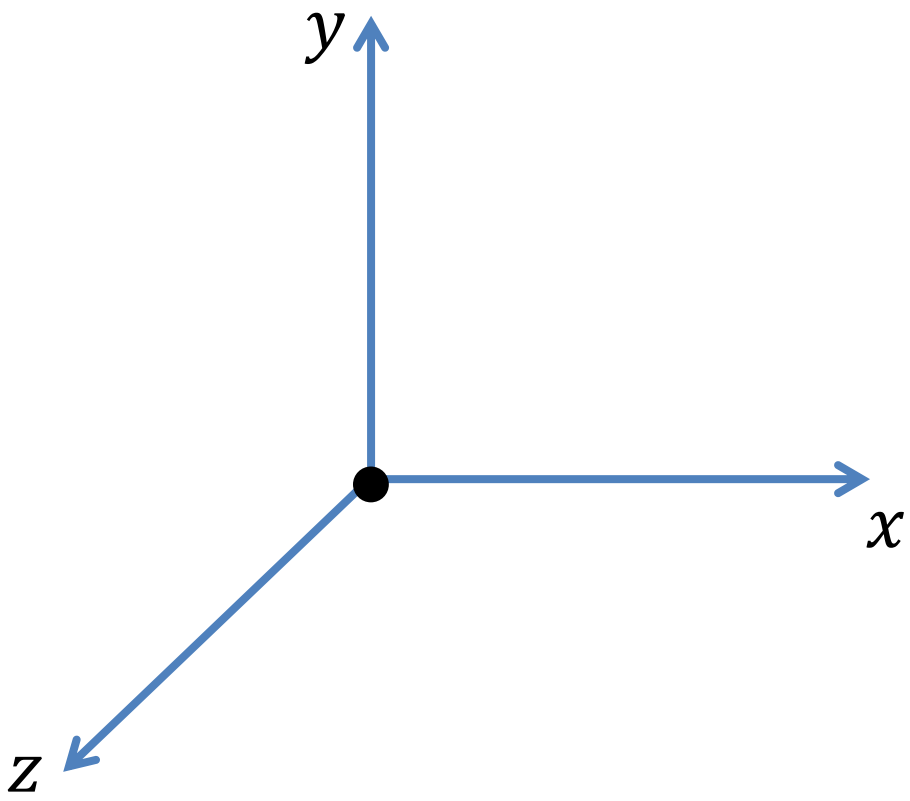
display → canvas

from visual import * → from vpython import *



Używając myszki można ruszać kamerą:
- prawy przycisk do obracania kamery
- dwa przyciski aby przybliżać/oddalać kamerę

Współrzędne w VPython



Kamera patrzy na (0,0,0)

Sfera

vp_first_prog_2.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_2.py

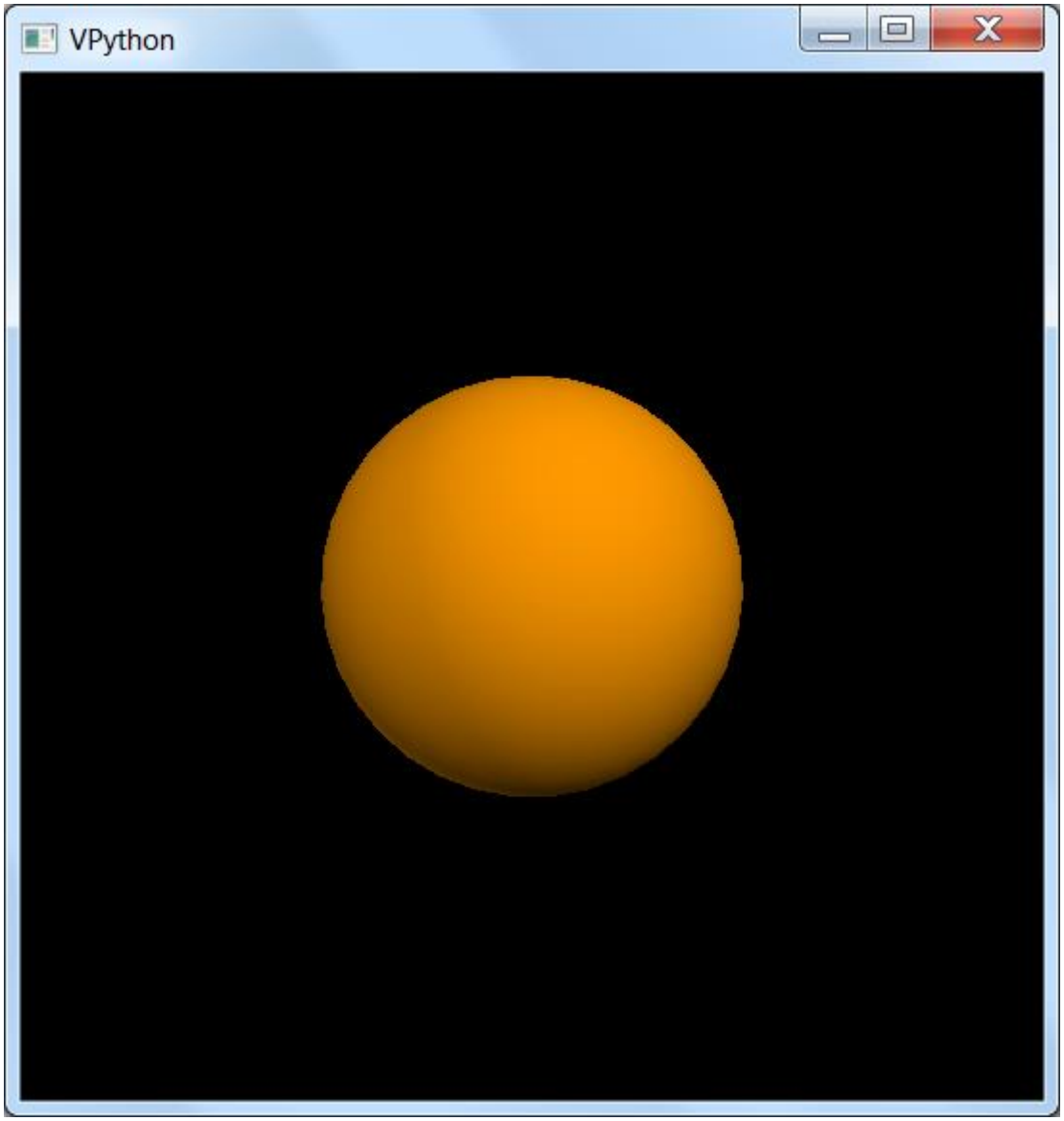
File Edit Format Run Options Windows Help

```
from visual import *

scene = display(width=550, height=580)
scene.range = 2

ball = sphere(pos=(0,0,0), radius=0.8, color=color.orange)

# color - red, green, blue, yellow, magenta, cyan,
#         orange, black, white

# color=(1,0.5,0.2) RGB
```

pos = pozycja (position)

Ln: 13 Col: 0

pos = vector(0,0,0) w VPython 7
lub
pos = vec(0,0,0)


VPython

sfera

vp_first_prog_3.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_3.py

File Edit Format Run Options Windows Help

```
from visual import *

ball = sphere(pos=(-2,1,0), radius=0.8, color=color.magenta)

print ball.pos
print ball.x
print ball.y
print ball.z
print ball.radius
print ball.color

ball.vel = vector(1,0,0)

print ball.vel
```

pos = vector(-2,1,0) w VPython 7

ball.x = ball.pos.x, ball.y = ball.pos.y

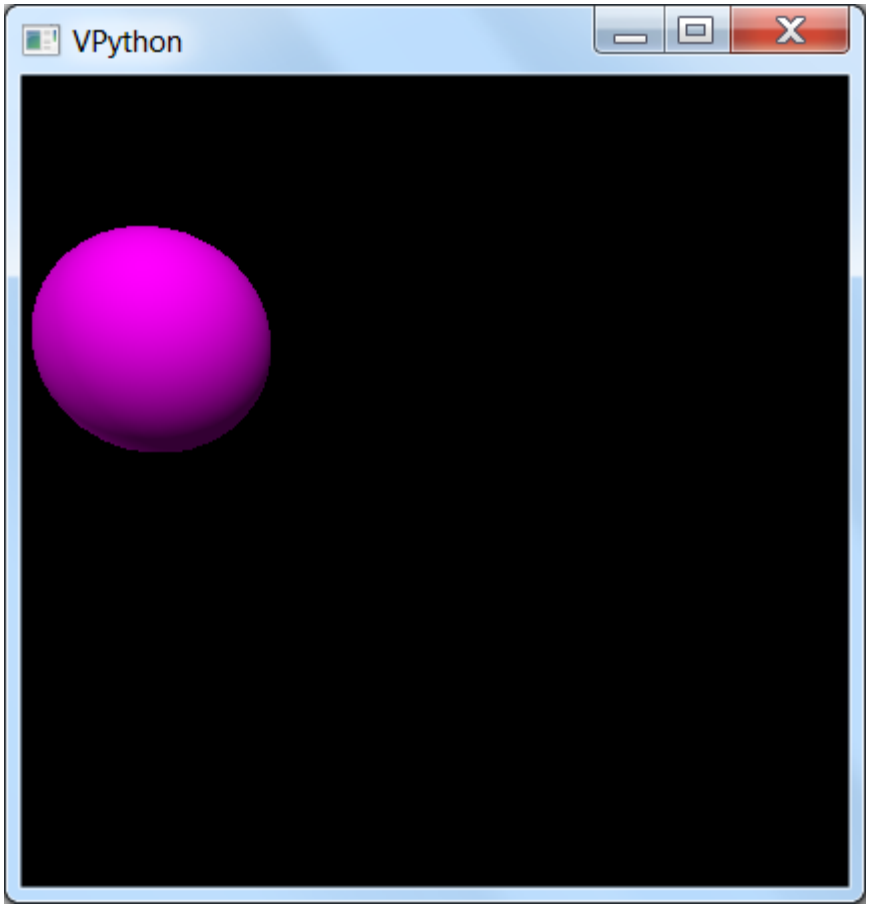
np., prędkość kuli

Ln: 16 Col: 0

```
<-2, 1, 0>
-2.0
1.0
0.0
0.8
(1.0, 0.0, 1.0)
<1, 0, 0>
```

w notacji RGB

Ln: 14 Col: 4


VPython

wektory

vp_first_prog_3b.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_3b.py

File Edit Format Run Options Windows Help

```
from visual import *

a = vector(2,0,0)
b = vector(0,1,4)

c = a + b

print c
print 2*c

print c.x
print c.y
print c.z

print mag(c)
```

długość wektora

Ln: 17 Col: 0

```
<2, 1, 4>
<4, 2, 8>
2.0
1.0
4.0
4.58257569496
```

Ln: 13 Col: 4

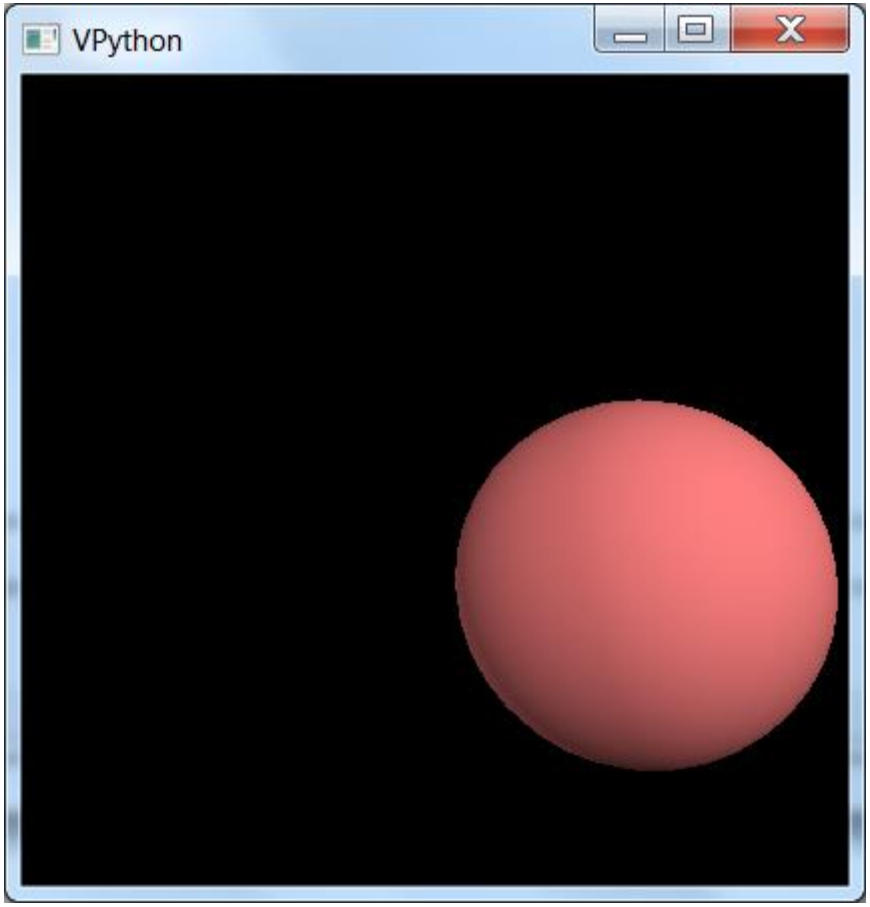
sfera

vp_first_prog_4.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_4.py

File Edit Format Run Options Windows Help

```
from visual import *

ball = sphere(pos=(-2,1,0), radius=0.8, color=color.magenta)

ball.x = 2
ball.y = -1
ball.radius = 1.8
ball.color = (1, 0.5, 0.5)   # RGB
```

Ln: 10 Col: 0


VPython

sfera, pudełko

vp_first_prog_5.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_5.py

File Edit Format Run Options Windows Help

```
from visual import *

scene = display(width=550, height=580)

ball = sphere(pos=(0,0,0), radius=0.8, color=color.red)

wallR = box(pos=(4,0,0), size=(0.1,7,7), color=color.cyan)
wallL = box(pos=(-4,0,0), size=(0.1,7,7), color=color.cyan)

print wallR.pos, wallR.size
```

Ln: 12 Col: 0

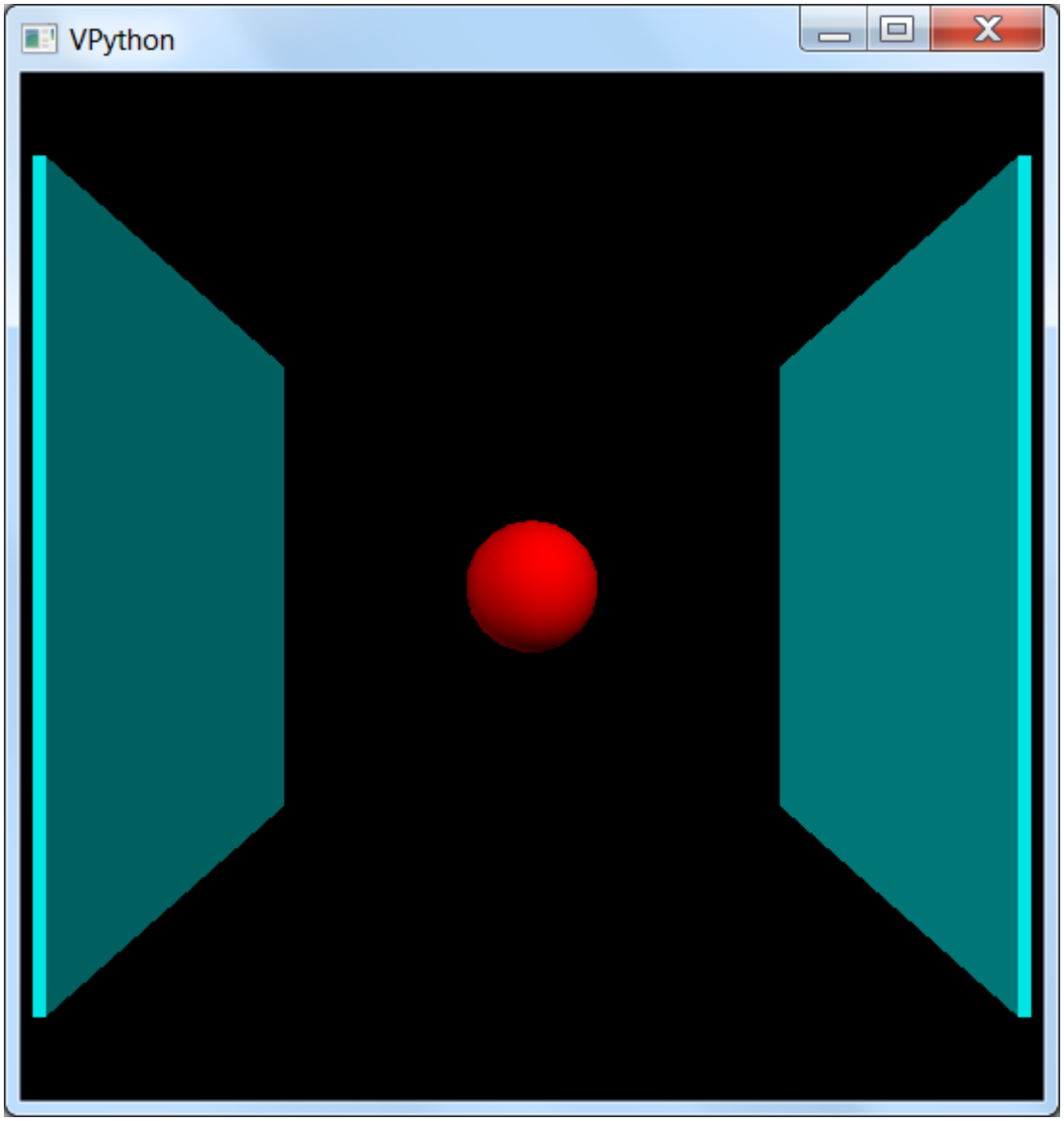
```
>>>
<4, 0, 0> <0.1, 7, 7>
```

Ln: 6 Col: 0

box: pos jest pozycją środka

size jest w (x,y,z)

size = vector(0.1,7,7) w VPython 7


VPython

ruch

vp_first_prog_6.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_6.py

File Edit Format Run Options Windows Help

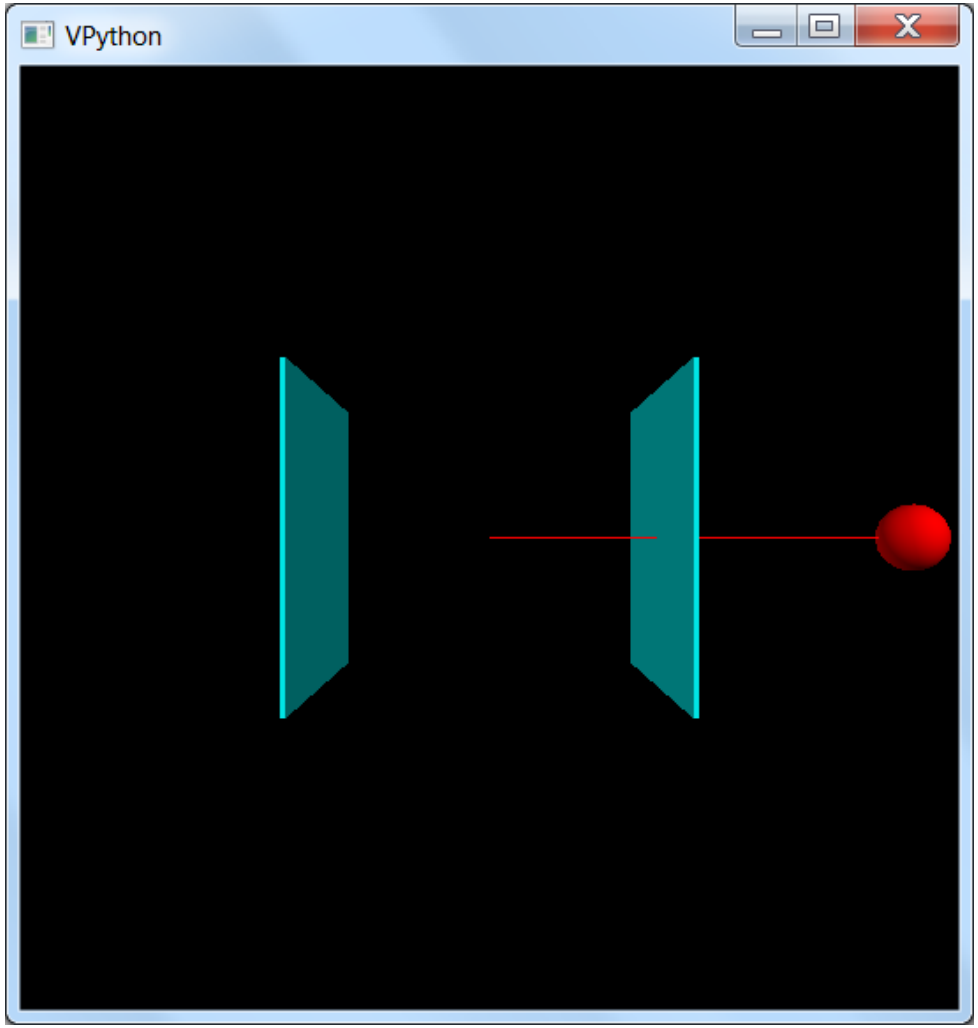
```
from visual import *

scene = display(width=550, height=580)

ball = sphere(pos=(0,0,0), radius=0.8, color=color.red,
              make_trail=True)

wallR = box(pos=(4,0,0), size=(0.1,7,7), color=color.cyan)
wallL = box(pos=(-4,0,0), size=(0.1,7,7), color=color.cyan)

ball.vel = vector(1,0,0)

t = 0
dt = 0.005

while t<10:
    rate(1000)
    ball.pos = ball.pos + ball.vel*dt
    t = t + dt
```

prędkość

musi być rate

Ln: 21 Col: 0

rate(N) – pętla nie będzie wykonana więcej niż N razy na sekundę



VPython oddala kamerę aby pokazać cała scenę (autoscaling)
Można wyłączyć
scene.autoscale = False
Po narysowaniu obiektów i przed pętlą

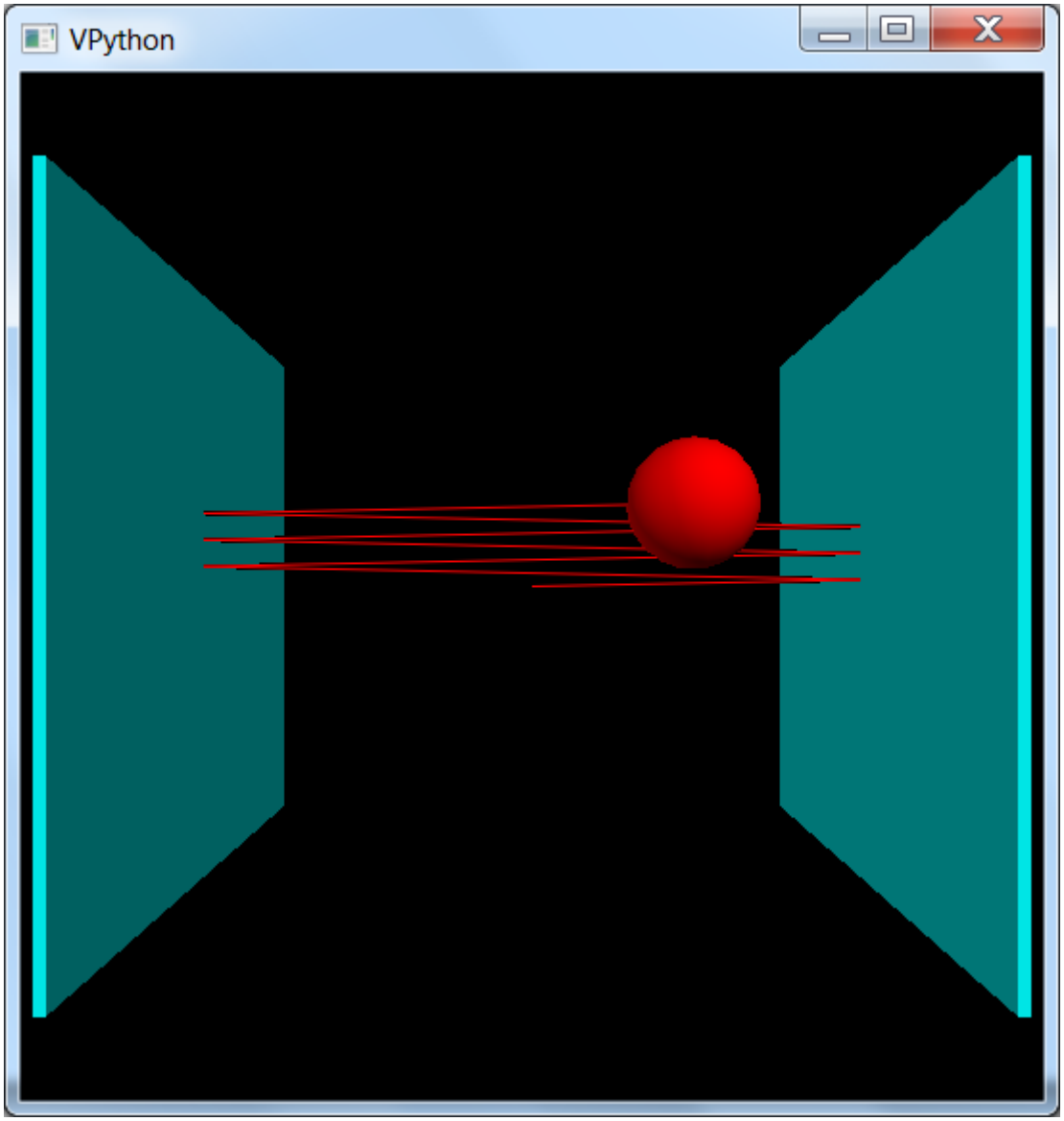
sfera, pudełko

vp_first_prog_7.py - C:\Users\Wysokie Energie\Desktop\VPython\lec_8\vp_first_prog_7.py

File Edit Format Run Options Windows Help

```
from visual import *

scene = display(width=550, height=580)

ball = sphere(pos=(0,0,0), radius=0.8, color=color.red,
              make_trail=True)

wallR = box(pos=(4,0,0), size=(0.1,7,7), color=color.cyan)
wallL = box(pos=(-4,0,0), size=(0.1,7,7), color=color.cyan)

ball.vel = vector(1, 0.02, 0)

t = 0
dt = 0.005

while t<50:
    rate(1000)
    ball.pos = ball.pos + ball.vel*dt
    if abs(ball.pos.x) >= wallR.pos.x:
        ball.vel.x = -ball.vel.x
    t = t + dt
```

Ln: 24 Col: 0


VPython