

10 O. Znajdź, z dokładnością do czterech cyfr dziesiętnych, wartości współczynników wielomianu interpolacyjnego opartego na następującej tabelce:

x	0.062500	0.187500	0.312500	0.437500	0.562500	0.687500	0.812500	0.935700
$f(x)$	0.687959	0.073443	-0.517558	-1.077264	-1.600455	-2.080815	-2.507266	-2.860307

Sporządź wykres uzyskanego wielomianu w przedziale $-1 \leq x \leq 1$ i zaznacz na nim punkty, które posłużyły do jego konstrukcji.

Zadanie rozwiązane poprzez interpolację za pomocą wielomianów. Nasza tabela wyznacza wielomian stopnia $n-1$. Naszym zadaniem jest wyznaczenie współczynników tego wielomianu, które łatwo otrzymamy podstawiając do wielomianu ogólnego $n-1$ stopnia informacji z tabeli co stworzy nam układ równań, który zapisany w postaci macierzowej (macierz Vandermonde'a). Układ ten został rozwiązany za pomocą faktoryzacji LU z biblioteki `numpy.linalg`. Następnie został narysowany wykres funkcji interpolowanej na danym przedziale.

```
import numpy as np
from numpy import power as p
import matplotlib.pyplot as plt

n=8 #wymiar macierzy
x = np.array([0.062500,0.187500,0.312500,0.437500,0.562500,0.687500,0.812500,0.935700])
y = np.array([0.687959,0.073443,-0.517558,-1.077264,-1.600455,-2.080815,-2.507266,-2.860307])

x_new=np.linspace(-1, 1)

size=x.size
Vandermond=np.zeros((size,size))
b=y.copy()

for i in range(0,n):
    for j in range(0,n):
        Vandermond[i,j]=p(x[i],n-1-j)

solves=np.linalg.solve(Vandermond,b)
print("Solves:")
for i in solves:
    print(i)

polynomial=0
for i in range(0,n):
    polynomial+=solves[i]*p(x_new,n-1-i)

plt.plot(x,y,'o',color='#000000')
plt.plot(x_new,polynomial)
plt.show()
```

Rozwiązania:

-1.92066440767
5.88567563737
-6.30988553487
2.72060536065
0.313038231168
0.326209509658
-5.03041324635
1.00097330742

Wykres funkcji interpolowanej