静岡理工科大学　情報学部　幸谷担当講義

レポート用紙

講義名　：数値解析１	年月日：　2024 年　　6 月　　3 日(月)
学籍番号：99999999	氏名：幸谷　智紀

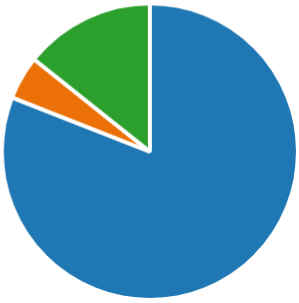
本日の課題　P.30　問題 4.3, P.32　問題 4.4

感想

2. 課題の難易度はどうでしたか？

詳細

難しかった	17
少し難しかった	1
ちょうど良かった	3
簡単だった	0
とても簡単だった	0



問題 4.3

プログラム例：

```
# num_diff2.py: 数値微分, 問題 4.3
import numpy as np # NumPy

# 元の関数
def func(x):
    # return np.exp(np.cos(x)) - x ** 3
    return np.sin(np.exp(x))

# 真の導関数
def true_dfunc(x):
    # return -np.exp(np.cos(x)) * np.sin(x) - 3 * x ** 2
    return np.cos(np.exp(x)) * np.exp(x)

# 前進差分商: (f(x + h) - f(x)) / h
def forward_diff(x, h, f):
    return (f(x + h) - f(x)) / h

# 中心差分商: (f(x + h) - f(x - h)) / 2h
def central_diff(x, h, f):
```

```
    return (f(x + h) - f(x - h)) / (2.0 * h)

# 後退差分商: (f(x) - f(x - h)) / h
def backward_diff(x, h, f):
    return (f(x) - f(x - h)) / h

# x = [-5, 5]
x = np.linspace(-5, 5, 4)
h = 10.0**(-5)

  # 前進差分商，中心差分商，後退差分商
  fdiff = forward_diff(x, h, func)
  cdiff = central_diff(x, h, func)
  bdiff = backward_diff(x, h, func)

  # 相対誤差チェック
  print('x                  = ', x)
  reldiff = np.abs((fdiff - true_dfunc(x)) / true_dfunc(x))
  print('Forward diff relerr = ', reldiff)
  reldiff = np.abs((cdiff - true_dfunc(x)) / true_dfunc(x))
  print('Central diff relerr = ', reldiff)
  reldiff = np.abs((bdiff - true_dfunc(x)) / true_dfunc(x))
  print('Backword diff relerr= ', reldiff)
```

実行結果：

```
x                   =  [-5.         -1.66666667  1.66666667  5.        ]
Forward diff relerr =  [4.99975922e-06 4.81950639e-06 4.52209204e-05 6.98280479e-04]
Central diff relerr =  [1.70121638e-11 2.19716179e-11 4.11010555e-11 3.74154823e-07]
Backword diff relerr=  [4.99979324e-06 4.81946244e-06 4.52210026e-05 6.97532169e-04]
```

問題 4.4

プログラム例：

```
  # roots_poly2.py : 代数方程式を解く 問題 4.4
  import numpy as np # NumPy
  import numpy.polynomial.polynomial as npppoly # Polynomial モジュール

  # 多項式の係数
  # 1. 学籍番号 1823054
  poly_coef = [4, 5, 0, 3, 2, 8, 1] # 4 + 5x + 0x^2 + 3x^3 + 2x^4 + 8x^5 +
```

```
1x^6
print('1. coef = ', poly_coef)
# 代数方程式の解(根)を導出
approx_roots = npppoly.polyroots(poly_coef)
approx_roots.sort() # 並び替え
print('approx_roots = ', approx_roots)

# 検算 p(x) == 0 ?
print('p(x) = ', npppoly.polyval(approx_roots, poly_coef))

# 2. [1, 1, ..., 1]
for n in [1, 2, 5, 10]:
    poly_coef = [1 for i in range(n + 1)]
    print('2. n, coef = ', n, poly_coef)

    # 代数方程式の解(根)を導出
    approx_roots = npppoly.polyroots(poly_coef)
    approx_roots.sort() # 並び替え
    print('approx_roots = ', approx_roots)

    # 検算 p(x) == 0 ?
    print('p(x) = ', npppoly.polyval(approx_roots, poly_coef))
```

実行結果：

レポート用紙

```
1. coef =  [4, 5, 0, 3, 2, 8, 1]
approx_roots =  [-7.79399373+0.j         -0.60317115+0.j         -0.46018778-0.86690694j
 -0.46018778+0.86690694j  0.65877022-0.6702986j   0.65877022+0.6702986j ]
p(x) =  [ 8.18531909e-11+0.00000000e+00j -1.77635684e-15+0.00000000e+00j
 -8.88178420e-16-7.10542736e-15j -8.88178420e-16+7.10542736e-15j
 -5.59552404e-14+3.93018951e-14j -5.59552404e-14-3.93018951e-14j]
2. n, coef =  1 [1, 1]
approx_roots =  [-1.]
p(x) =  [0.]
2. n, coef =  2 [1, 1, 1]
approx_roots =  [-0.5-0.8660254j -0.5+0.8660254j]
p(x) =  [1.11022302e-16-5.55111512e-17j 1.11022302e-16+5.55111512e-17j]
2. n, coef =  5 [1, 1, 1, 1, 1, 1]
approx_roots =  [-1. +0.j        -0.5-0.8660254j -0.5+0.8660254j  0.5-0.8660254j
  0.5+0.8660254j]
p(x) =  [ 1.66533454e-15+0.00000000e+00j  6.66133815e-16-5.55111512e-17j
  6.66133815e-16+5.55111512e-17j -1.11022302e-15+9.99200722e-16j
 -1.11022302e-15-9.99200722e-16j]
2. n, coef =  10 [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
approx_roots =  [-0.95949297-0.28173256j -0.95949297+0.28173256j -0.65486073-0.75574957j
 -0.65486073+0.75574957j -0.14231484-0.98982144j -0.14231484+0.98982144j
  0.41541501-0.909632j    0.41541501+0.909632j    0.84125353-0.54064082j
  0.84125353+0.54064082j]
p(x) =  [ 0.00000000e+00-1.72084569e-15j  0.00000000e+00+1.72084569e-15j
  3.99680289e-15-1.27675648e-15j  3.99680289e-15+1.27675648e-15j
 -1.99840144e-15+3.88578059e-15j -1.99840144e-15-3.88578059e-15j
  1.11022302e-14-1.93178806e-14j  1.11022302e-14+1.93178806e-14j
 -1.08801856e-14+1.42108547e-14j -1.08801856e-14-1.42108547e-14j]
```

以上