

計算物理学 II 第 2 回レポート課題

提出期限：2025 年 11 月 28 日

以下の課題 1, 2, 3, 4 に取り組み, その結果を \LaTeX でレポートにまとめよ. なお, 以下の点に留意せよ.

- レポートにはタイトルを付け, 氏名, 学籍番号, 所属, レポート作成日を記載すること.
- レポート作成時に, この.pdf ファイルのソースファイル (lecture6/report2/main.tex) を活用しても構わない.
- 読みやすく, 体裁の整ったレポート作成を心がけて欲しい.
- 作成したレポートの.pdf ファイルと課題 1, 課題 2, 課題 4 で使用したソースコード (.py ファイル) を manaba に提出すること.

1

Leibniz (ライプニッツ) の公式,

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = \frac{\pi}{4} \quad (1.1)$$

を使って, π の近似値を次のように得ることを考える.

$$\pi \simeq 4 \sum_{n=0}^N \frac{(-1)^n}{2n+1}. \quad (1.2)$$

$N = 10^5$ の時の π の近似値を求めよ.

2

Wallis (ウォリス) の公式

$$\prod_{n=1}^{\infty} \left(\frac{2n}{2n-1} \cdot \frac{2n}{2n+1} \right) = \frac{\pi}{2} \quad (2.1)$$

を使って, π の近似値を次のように得ることを考える.

$$\pi \simeq 2 \prod_{n=1}^N \left(\frac{2n}{2n-1} \cdot \frac{2n}{2n+1} \right). \quad (2.2)$$

$N = 10^5$ の時の π の近似値を求めよ.

3

課題 1, 2 で求めた $N = 10^5$ での π の近似値について、どちらの方が真の π に近い結果を与えているか調べよ.*¹

4

$f(x)$ および $g(x)$ として、異なる初等関数を自由に二つ選び、以下の課題に取り組んでみよ。

- (1) 自分の選んだ初等関数を明記した上で、 $f(x)$, $g(x)$ の Taylor 展開を書け。
- (2) 以下のように、次数 N で打ち切られた $f(x)$ の Taylor 展開を使って $f(a)$ の近似値を求めることを考える。ただし、 a は $f(x)$ の Taylor 展開の収束半径内にある数値とし、自由に選んでよい。
 - 与えられた a と N に対して、次数 N で打ち切られた Taylor 展開から $f(a)$ の近似値を求めるプログラムを作成すること。例えば、ソースコード 1 は、 $f(x) = e^x$ の場合のサンプルプログラムである。20 行目以下では、定義した関数 `taylor_exp` の引数を `for` 文で回すことで、 $N = 1, 2, \dots, 10$ までの結果が得られるようになっている。
 - Python で初等関数を使う場合、`math` モジュールにそれらが用意されている。`math` モジュールで使える数学関数については、<https://docs.python.org/ja/3/library/math.html> を参照するとよい。
 - 「`LATEX` 入門」の 5 節、あるいはこの.pdf ファイルのソースファイルを参照し、得られた結果を表 1 のようにまとめよ。ただし、下の行に行くほど打ち切り次数 N の値が大きくなるように並べること。なお、 N の値は好きに選べばよいが、最低でも三つは選んで表に載せること。また、一番下の行には $N = \infty$ の場合、すなわち $f(a)$ の厳密な値を記載せよ。
- (3) $g(x)$ についても問 (2) と同様のことを行え.*²

表 1 $f(a)$ の Taylor 展開による近似

打ち切り次数 N	$f(a)$ の近似値
2	得られた近似値
4	得られた近似値
6	得られた近似値
8	得られた近似値
10	得られた近似値
∞	厳密な値

*¹ `lecture_material_5.pdf` の 67 ページ参照。

*² すなわち、次数 N で打ち切られた $g(x)$ の Taylor 展開を使って $g(b)$ の近似値を求め、厳密な値へ近づいていくことを確認せよ。ただし、 b は $g(x)$ の Taylor 展開の収束半径内にある数値とし、自由に選んでよい。

```
1 import math
2
3 def taylor_exp(value_x,num_terms):
4     acc = 0
5     num = 1
6     den = 1
7
8     for index in range(num_terms):
9         nextTerm = num / den
10        acc = acc + nextTerm
11        num = num * value_x
12        den = den * (index+1)
13
14    return acc
15
16 value_x = 0.1
17 exact = math.exp(value_x)
18 print("Exact = ",exact)
19
20 for num_terms in range(1,10):
21     approximation = taylor_exp(value_x,num_terms)
22     print(num_terms,approximation,math.fabs(exact-approximation))
```

アンケート

この項目は成績とは無関係です。こちら（[クリック](#)）の Google フォームよりアンケートにご協力ください。