応用複素関数レポート課題3

桂田 祐史

2022 年 7 月 5 日, 2022 年 7 月 5 日

- 〆切は 7 月 30 日 (土曜)23:00 です。
- 提出先は Oh-o! Meiji のレポート・システム、形式は A4 サイズ PDF。
  もし容量制限 (1 ファイル 30MB) に引っかかった場合は、分割して送って下さい。
- 使用するプログラミング言語は、自分の MacBook で実行して見せることが可能なものであればなんでも可。
- プログラムとその実行結果、実行するための情報を含めること。
- 実行結果は、数表・グラフを適切に選択して分かりやすく提示すること。
  - 誤差などは固定小数点形式 (C 言語の `%f`) よりは指数形式 (C 言語の `%e`) を使う、むやみに多くの桁を表示しない、あるいは表よりはグラフ (対数目盛りが適当な場合も多い) を使う。
  - 逆に必要があれば (意味があるならば) 多くの桁数を表示させる (`%`$m.n$`f` などを使う。これは全体の幅 m 桁、小数点以下 n 桁という意味。)。
  - 書式の指定については、C 言語の場合は、例えば「浮動小数点数の入出力と四則演算」[1] の「書式指定ミニマム」を見よ。Python の場合も C 言語とほぼ同様の書式指定が出来る。例えばC 言語で

    ```
    printf("π=%20.15f, e=%e\n", pi, e);
    ```

    とするのと同じことが、Python では以下のようにして実現できる。

    ```
    print('π=%20.15f, e=%e' % (pi, e))
    ```

    で実現できる。
  - グラフ作成に Excel を使う人が例年少なくないけれど、C 言語で計算するならば gnuplot, Python で計算するならば matplotlib の機能を使うことを勧めます。
- 念のため: 授業で公開したサンプル・プログラムの入手法

C 言語 (次のようにして `prog20220628` というフォルダが現れる)

```
curl -O http://nalab.mind.meiji.ac.jp/~mk/complex2/prog20220628.tar.gz
tar xzf prog20220628.tar.gz
```

Python (Jupyter notebook 用)

```
curl -O http://nalab.mind.meiji.ac.jp/~mk/complex2/20220628ni.ipynb
```

[1] http://nalab.mind.meiji.ac.jp/~mk/labo/text/cminimum/node10.html

課題 3

次の (1)〜(5) からいずれか 1 つ選んでレポートせよ。

(1) 計算が困難であると予想される定積分 (初等関数で表せない、積分区間全体で被積分関数が滑らかではない) を自分で選び、数値積分で値を求める。その値がどれくらいの精度か (誤差がどの程度か)、何らかの方法で確認すること。なるべく複数の方法で計算すること。分割を細かくすると精度がどのように変わるか調べること。

(2) Euler のガンマ定数 γ は、普通 $\gamma := \lim_{n\to\infty}\left(\sum_{k=1}^{n}\frac{1}{k} - \log n\right)$ で定義されるが、この式で γ の値を計算するのは難しい。

$$\gamma = -\int_0^1 \log\log\frac{1}{x}\,dx \tag{1}$$

が成り立つことが知られている。この右辺を DE 公式で数値積分することで γ の近似値を求めよ。被積分関数 $f(x) = -\log\log\frac{1}{x}$ がどういう関数か調べること。結果を何らかの方法でチェックすること (誤差がどの程度か)。もし出来れば、(1) がなぜ成り立つか調べること。

(3) ガンマ関数 $\Gamma(x) := \int_0^\infty e^{-t}t^{x-1}\,dt$ $(x > 0)$ を数値積分することにより計算するプログラムを作り、どういう範囲の x に対して、どの程度の精度が得られるか、調べよ。被積分関数 $e^{-t}t^{x-1}$ がどのような関数か、理解した上で取り組むこと。

(注) よく知られている関数等式 $\Gamma(x) = (x-1)\Gamma(x-1)$ を利用すると、どこか都合の良い幅 1 の区間に属する x に対して数値積分で $\Gamma(x)$ を求めれば良いことになる。

(4) $I = \int_a^b f(x)\,dx$ に対する数値積分公式では、f の値のみ用い、f の導関数の値は使わないのが普通であるが、f' の値を使って良いならば、**補正台形公式**と呼ばれる

$$T_{N,\text{補}} := T_N - \frac{h^2}{12}\left(f'(b) - f'(a)\right)$$

が利用できる。台形公式 T_N と比べて、$T_{N,\text{補}}$ では精度がどれくらい改善されるか、適当な被積分関数を選んで実験して調べよ。さらに次のどちらかを行うこと。

(a) Euler-Maclaurin の定理 (7 月 5 日の授業スライドで紹介してある) を参考にして、さらに高次の補正を試みる。

(b) 中点公式 M_N で同様の補正をする。

(5) 講義で説明した関数 $f(x) = \dfrac{1}{1+25x^2}$ 以外の関数に対して、Runge の現象が起こるかどうか調べよ (本質的に違うものを複数選んで実験すること)。