**Tugas 3 Praktikum Metode Numerik**

**Semester Genap Tahun Ajaran 2022/2023**

**Petunjuk Umum:**

1. Kerjakan secara individu.
2. Kerjakan tugas ini menggunakan bahasa pemrograman Python dengan *file format* berupa *interactive Python notebook* (yaitu *file* berbentuk **.ipynb** BUKAN .py), yang bisa dibuat misalnya menggunakan Jupyter Notebook atau Google Colaboratory.
3. **Harap sertakan penjelasan untuk setiap variabel yang digunakan dan setiap proses secara singkat** di samping potongan kode (dengan *comment*, ‘#'). Selain itu, **sertakan juga penjelasan program** (yang bisa mencakupi idenya apa, bagaimana cara eksekusi program, atau tentang algoritma program yang digunakan) **pada *cell* di sebelah (atas/bawah) program**.
4. Format nama *file* untuk Tugas 3 adalah:

**Nama Lengkap\_NPM\_Kelas SIAK\_Tugas3PrakMetnum.ipynb**

Contoh penamaan yang benar:

Johan Frederik Steffensen\_2234567890\_Kelas F\_Tugas3PrakMetnum.ipynb

1. Pengumpulan tugas dilakukan ke Google Forms berikut ini, sesuai dengan kelas Anda di SIAK NG (*link* akan selalu sama untuk semua tugas praktikum metode numerik):

* Kelas A: <https://forms.gle/Wgin1RdVyrNZ7xDbA>
* Kelas B: <https://forms.gle/cqurG75aqdTLq1LR8>
* Kelas C: <https://forms.gle/yPKk42n1mTLrJzFE6>
* Kelas D: <https://forms.gle/7TnSmCT9DcmhBvvo6>
* Kelas E: <https://forms.gle/SSQUksLtwsLkYTxT6>

1. Durasi pengerjaan Tugas 3 ini adalah **2 (dua) minggu**, dan tenggat waktu (*deadline*) pengumpulan Tugas 3 ini adalah

**Minggu, 30 April 2023, pukul 23.59 WIB.**

Mohon manfaatkan waktu Anda dengan baik (seperti mencicil pengerjaan, bahkan sudah selesai dari jauh-jauh hari) agar mengumpulkan tugas sebelum *deadline*. Keterlambatan pengumpulan akan dikenakan pengurangan nilai (sudah dibicarakan dengan dosen). Lebih baik tidak terlambat daripada terlambat, dan lebih baik terlambat daripada tidak mengumpulkan sama sekali.

1. Sesuai standar Universitas Indonesia, **plagiarisme dilarang keras** dan bisa menyebabkan nilai tugas praktikum **menjadi nol untuk semua pihak yang terlibat, tanpa peringatan apapun.**
2. *Module* atau *package* Python yang boleh digunakan (di-*import*) untuk Tugas 3 ini hanyalah numpy dan tabulate. Apabila Anda berniat ingin menggunakan *module* lain, harap konfirmasikan ke narahubung terlebih dahulu (bisa saja diperbolehkan).
3. Narahubung untuk Tugas 3 Praktikum Metode Numerik adalah:

Bisma Rohpanca Joyosumarto (ID LINE: bisma\_joyosumarto)

Silakan hubungi narahubung di atas apabila ada yang ingin ditanyakan atau dikonfirmasikan.

**Soal Tugas 3 Praktikum Metode Numerik**

Sebelum memulai pengerjaan tugas ini, harap hitung nilai k berikut ini berdasarkan NPM Anda:

**k = 1 + ((NPM) mod 4)**

Diberikan fungsi :

**Soal bagian wajib:**

Buatlah program untuk menentukan nilai pada secara numerik, dengan ketentuan sebagai berikut.

1. Gunakan kelima metode diferensiasi numerik berikut ini, dengan :
2. *Forward-Difference*
3. *Three-Point Endpoint* (TPEP)
4. *Three-Point Midpoint* (TPMP)
5. *Five-Point Endpoint* (FPEP)
6. *Five-Point Midpoint* (FPMP)
7. Dengan nilai h awal (bukan 0.01), terapkan ekstrapolasi Richardson pada metode…

* TPEP, jika k = 1;
* TPMP, jika k = 2;
* FPEP, jika k = 3; atau
* FPMP, jika k = 4.

Cara menerapkannya: buatlah tabel dengan data , , , dan yang diperoleh dari metode diferensiasi numerik tersebut.

**Bagian bonus** (tidak wajib; hanya untuk menaikkan nilai apabila jawaban untuk bagian wajib belum sempurna):

* Program juga menghitung aproksimasi pada dengan menggunakan metode *Second Derivative Midpoint* (SDMP).
* Program bisa menerima input berupa sembarang fungsi dan sembarang nilai h (baik untuk bagian diferensiasi numerik maupun ekstrapolasi Richardson).
* Program bisa *looping*, menerima input berkali-kali sesuai permintaan pengguna, tanpa harus di-*run* ulang terlebih dahulu.
* Keseluruhan program dikemas di dalam satu *function*.

**Catatan:**

* Untuk penerapan ekstrapolasi Richardson pada FPEP dan FPMP, nama kolom pertama tabel perlu bertuliskan O(h^4), diikuti kolom berikutnya yaitu O(h^6), O(h^8), O(h^10).
* Fungsi ln dari numpy bernama log.

**Contoh *output* program (jika k = 1)**

| Program ini akan mengaproksimasi nilai turunan f(x)  dengan metode diferensiasi numerik dan ekstrapolasi Richardson.  NPM saya adalah 2106635580 sehingga k = 1.  Masukkan rumus f(x),  boleh bergantung k: x \* arctan(x - k) - log(sqrt(x\*\*2 + 1)) + (1+sin(x))\*\*(k\*x)  Masukkan nilai x: pi  Masukkan nilai h untuk diferensiasi numerik: 0.01  Masukkan nilai h untuk ekstrapolasi Richardson: 0.1  Turunan pertama dari f di x0 = 3.141592653589793 adalah :  f'(3.141592653589793) = -1.7104768791706526 (Forward/Backward-Difference)  f'(3.141592653589793) = -1.7346748612307117 (Three Point Endpoint)  f'(3.141592653589793) = -1.734123961062961 (Three Point Midpoint)  f'(3.141592653589793) = -1.7343125132107675 (Five Point Endpoint)  f'(3.141592653589793) = -1.7343126641320576 (Five Point Midpoint)  Turunan kedua dari f di x0 = 3.141592653589793 adalah :  f''(3.141592653589793) = 4.729416378461693 (Second Derivative Midpoint)  Karena k = 1, ekstrapolasi Richardson akan diterapkan pada  metode Three-Point Endpoint (TPEP).  Tabel Ekstrapolasi Richardson untuk O(h^2j)  | O(h^2) | O(h^4) | O(h^6) | O(h^8) |  |----------+---------------------+---------------------+---------------------|  | -1.75601 | | | |  | -1.74178 | -1.7370386500294226 | | |  | -1.73643 | -1.7346459339256375 | -1.7344864195187184 | |  | -1.73487 | -1.7343534703621828 | -1.7343339727912859 | -1.7343315530019616 |  Apakah Anda ingin mengulang program? (y/n): n  Terima kasih telah menggunakan program. |
| --- |

**Contoh *output* program (jika k = 2)**

| Program ini akan mengaproksimasi nilai turunan f(x)  dengan metode diferensiasi numerik dan ekstrapolasi Richardson.  NPM saya adalah 2106635581 sehingga k = 2.  Masukkan rumus f(x),  boleh bergantung k: x \* arctan(x - k) - log(sqrt(x\*\*2 + 1)) + (1+sin(x))\*\*(k\*x)  Masukkan nilai x: pi  Masukkan nilai h untuk diferensiasi numerik: 0.01  Masukkan nilai h untuk ekstrapolasi Richardson: 0.1  Turunan pertama dari f di x0 = 3.141592653589793 adalah :  f'(3.141592653589793) = -4.213984924615 (Forward/Backward-Difference)  f'(3.141592653589793) = -4.35452558561038 (Three Point Endpoint)  f'(3.141592653589793) = -4.357906284450319 (Three Point Midpoint)  f'(3.141592653589793) = -4.356806519104639 (Five Point Endpoint)  f'(3.141592653589793) = -4.3568019567525065 (Five Point Midpoint)  Turunan kedua dari f di x0 = 3.141592653589793 adalah :  f''(3.141592653589793) = 28.7842719670639 (Second Derivative Midpoint)  Karena k = 2, ekstrapolasi Richardson akan diterapkan pada  metode Three-Point Midpoint (TPMP).  Tabel Ekstrapolasi Richardson untuk O(h^2j)  | O(h^2) | O(h^4) | O(h^6) | O(h^8) |  |----------+--------------------+--------------------+---------------------|  | -4.46508 | | | |  | -4.38429 | -4.35736744856919 | | |  | -4.3637 | -4.356836761563017 | -4.356801382429271 | |  | -4.35853 | -4.356803277613934 | -4.356801045350662 | -4.3568010400002075 |  Apakah Anda ingin mengulang program? (y/n): n  Terima kasih telah menggunakan program. |
| --- |

**Contoh *output* program (jika k = 3)**

| Program ini akan mengaproksimasi nilai turunan f(x)  dengan metode diferensiasi numerik dan ekstrapolasi Richardson.  NPM saya adalah 2106635582 sehingga k = 3.  Masukkan rumus f(x),  boleh bergantung k: x \* arctan(x - k) - log(sqrt(x\*\*2 + 1)) + (1+sin(x))\*\*(k\*x)  Masukkan nilai x: pi  Masukkan nilai h untuk diferensiasi numerik: 0.01  Masukkan nilai h untuk ekstrapolasi Richardson: 0.1  Turunan pertama dari f di x0 = 3.141592653589793 adalah :  f'(3.141592653589793) = -6.127440974910225 (Forward/Backward-Difference)  f'(3.141592653589793) = -6.4793757827599165 (Three Point Endpoint)  f'(3.141592653589793) = -6.5003966847775985 (Three Point Midpoint)  f'(3.141592653589793) = -6.493313955104585 (Five Point Endpoint)  f'(3.141592653589793) = -6.493302409810006 (Five Point Midpoint)  Turunan kedua dari f di x0 = 3.141592653589793 adalah :  f''(3.141592653589793) = 74.5911419734746 (Second Derivative Midpoint)  Karena k = 3, ekstrapolasi Richardson akan diterapkan pada  metode Five-Point Endpoint (FPEP).  Tabel Ekstrapolasi Richardson untuk O(h^2j)  | O(h^4) | O(h^6) | O(h^8) | O(h^10) |  |----------+--------------------+--------------------+-------------------|  | -6.43131 | | | |  | -6.49473 | -6.515873888170528 | | |  | -6.49367 | -6.493317029829442 | -6.49181323927337 | |  | -6.49333 | -6.493220173705536 | -6.493213716630609 | -6.49323594642993 |  Apakah Anda ingin mengulang program? (y/n): n  Terima kasih telah menggunakan program. |
| --- |

**Contoh *output* program (jika k = 4)**

| Program ini akan mengaproksimasi nilai turunan f(x)  dengan metode diferensiasi numerik dan ekstrapolasi Richardson.  NPM saya adalah 2106635583 sehingga k = 4.  Masukkan rumus f(x),  boleh bergantung k: x \* arctan(x - k) - log(sqrt(x\*\*2 + 1)) + (1+sin(x))\*\*(k\*x)  Masukkan nilai x: pi  Masukkan nilai h untuk diferensiasi numerik: 0.01  Masukkan nilai h untuk ekstrapolasi Richardson: 0.1  Turunan pertama dari f di x0 = 3.141592653589793 adalah :  f'(3.141592653589793) = -11.074348353757468 (Forward/Backward-Difference)  f'(3.141592653589793) = -11.716917940575655 (Three Point Endpoint)  f'(3.141592653589793) = -11.776486856555213 (Three Point Midpoint)  f'(3.141592653589793) = -11.755933962256709 (Five Point Endpoint)  f'(3.141592653589793) = -11.755971623158283 (Five Point Midpoint)  Turunan kedua dari f di x0 = 3.141592653589793 adalah :  f''(3.141592653589793) = 140.4277005595489 (Second Derivative Midpoint)  Karena k = 4, ekstrapolasi Richardson akan diterapkan pada  metode Five-Point Midpoint (FPMP).  Tabel Ekstrapolasi Richardson untuk O(h^2j)  | O(h^4) | O(h^6) | O(h^8) | O(h^10) |  |----------+---------------------+---------------------+---------------------|  | -11.7183 | | | |  | -11.7529 | -11.764389140607797 | | |  | -11.7558 | -11.756739622850658 | -11.756229655000181 | |  | -11.756 | -11.756028606319008 | -11.755981205216898 | -11.755977261569544 |  Apakah Anda ingin mengulang program? (y/n): n  Terima kasih telah menggunakan program. |
| --- |