

PENGUNAAN PROSEDUR REGULA-FALSI DAN INTERVAL BAGI DUA UNTUK PENCARIAN AKAR PADA FUNGSI POLINOMIAL BERDERAJAT TIGA

USE OF RULE-FALSI AND THE INTERVAL DIVIDE BY TWO TO FIND THE ROOTS OF A PLYNOMINAL FUNCTION OF THE THIRD DEGREE

Nur Hafidz ¹, Moh Rofi ², Adhi Priyanto ³

ABSTRACT

¹STMIK AKI PATI
Jln Kamandowo No. 13, Pati,
Jawa Tengah, Indonesia
nurhafid348gmail.com

²STMIK AKI PATI
Jln. Kamandowo No 13 Pati,
Jawa Tengah, Indonesia
mohrofi2017@gmail.com

³STMIK AKI PATI
Jln. Kamandowo No. 13 Pati
Jawa Tengah, Indonesia
adhi.stmikaki@gmail.com

Finding the root of a function cannot always be solved analytically, namely an equation of a non-linear function, so it must be done numerically with the help of a computer to perform computations, using a certain programming language. transcendental function equations and polynomial function equations. In the Non Linear Equation System, among several solution Procedures that are often used are the Newton-Raphson, the Secant, the Bisection (Dividing Interval), the False-Regulatory Procedure. In this case, the authors interested to analyzing two existing Procedures in finding the root of a non-linear function in the form of a three-degree polynomial function. This process of analysis uses the Java of programming language tool, with the reason to make it easier in terms of the algorithm analysis of completion of Procedure, and to make it easier in monitoring in terms of execution time and analysis of the output results. Recording to that reason, we can clearly know what the differences are between the False-Position Procedure and Interval Bisection Procedure.

Keywords : *False-Position Procedure, Interval Bisection Procedure, three-degree polynomial function*

1. PENDAHULUAN

Dalam dunia industri sering dijumpai permasalahan pencarian akar dari suatu fungsi riil yang bukan merupakan bilangan eksak, sehingga harus dicari dengan suatu prosedur numerik tertentu. Nilai x yang memenuhi persamaan suatu fungsi riil $f(x) = 0$ dikenal dengan akar fungsi riil $f(x)$ tersebut. Fungsi riil $f(x)$ yang akarnya bukan merupakan bilangan eksak umumnya berupa fungsi non linear, dengan ciri khas grafiknya berupa garis bukan lurus.

Suatu fungsi riil non linear misal $f(x)$ dapat dikelompokkan menjadi dua, yakni fungsi transendental dan fungsi polinomial. Fungsi Transendental merupakan suatu fungsi riil yang memuat sejumlah variabel dan konstanta, yang tidak dapat diekspresikan dengan urutan berhingga dari operasi penjumlahan, pengurangan, maupun perkalian, sedangkan Fungsi Polinomial dapat diekspresikan dengan suatu $f(x)$, dengan persamaan.

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0, \quad (1)$$

dimana n merupakan derajat fungsi polinomial tersebut.

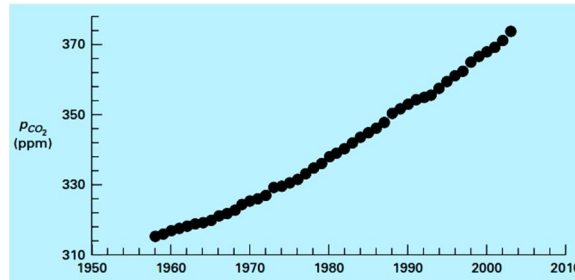
Penelitian yang dilakukan ini dimaksudkan untuk mengetahui tingkat akselerasi hasil iterasi pencarian akar dari suatu fungsi polinomial berderajat tiga, dengan prosedur Interval Bisection dan prosedur False-Position. Iterasi merupakan pengulangan langkah perhitungan yang dapat diperankan oleh suatu personal computer dan sejenisnya dengan suatu bahasa pemrograman. Untuk menampilkan hasil komputasi (iterasi pencarian akar) pada penelitian ini dipilih bahasa Java.

Salah satu penerapan pencarian akar suatu fungsi polinomial di bidang teknik yang pernah dilakukan adalah perhitungan tekanan parsial gas karbon dioksida (CO_2) yang terjadi di Mauna Loa,

Hawai pada tahun 1958 sampai 2003. Fenomena yang terjadi adalah adanya tren positif dari data fungsi tekanan parsial dari karbon dioksida (CO_2) di atmosfer, yang mengikuti persamaan fungsi $p\text{CO}_2$ terhadap tahun ().

$$p\text{CO}_2 = 0,011825 (t - 1980,5)^2 + 1,356975 (t - 1980,5) + 339$$

dimana $p\text{CO}_2$ = tekanan parsial dari karbon dioksida (CO_2) di atmosfer (ppm). Data yang diperoleh menunjukkan bahwa telah terjadi peningkatan lebih dari 19 % atau pada interval 315 – 376 ppm, pada rentang interval antara tahun 1958 sampai dengan 2003 ^[1]



Gambar 1. Rata-rata Tahunan Tekanan Parsial dari Gas Karbon Dioksida di Mauna Loa, Hawai

Penelitian sebelumnya yang pernah dilakukan antara lain adalah oleh Wulan pada tahun 2017, dalam rangka pencarian akar persamaan satu fungsi non linear dengan membandingkan penggunaan prosedur Newton-Raphson dengan prosedur Regula-Falsi, yang hasilnya adalah apabila menggunakan prosedur Newton-Raphson diperlukan 6 (enam) langkah iterasi, sedangkan apabila menggunakan prosedur Regula-Falsi diperlukan 8 (delapan) langkah iterasi.

Penelitian berikutnya dilakukan oleh Patrisius Batarius pada tahun 2018, yakni membandingkan hasil iterasi pencarian akar ganda dengan prosedur Newton-Raphson termodifikasi dengan prosedur Secant termodifikasi, dihasilkan untuk penggunaan prosedur Newton-Raphson termodifikasi lebih efektif, yang berarti jumlah langkah iterasinya lebih sedikit.

Penelitian selanjutnya dilakukan oleh Endang Sunandar dan Indrianto pada tahun 2020, yang melakukan pengujian perbedaan laju iterasi dengan menggunakan prosedur Newton-Raphson dan prosedur Secant untuk pencarian akar suatu persamaan fungsi non linear, yang menghasilkan laju iterasi Newton-Raphson sebanyak 6 (enam) langkah, dan laju iterasi dengan menggunakan prosedur Secant sebanyak 8 (delapan) langkah ^[2].

2. KAJIAN PUSTAKA

Iterasi Numerik dengan Prosedur Bagi-Dua dan Regula-Falsi

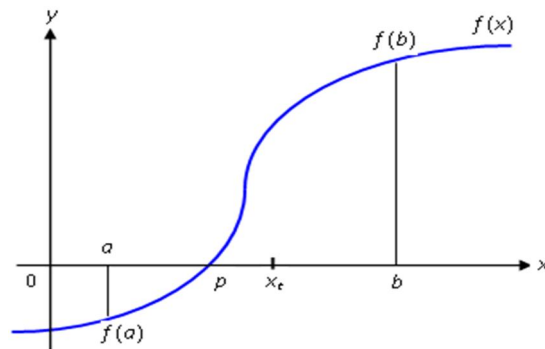
Dalam pencarian akar suatu persamaan fungsi $f(x)$, khususnya fungsi non linear dalam bentuk polinomial diperlukan prosedur aproksimasi (pendekatan) secara numerik. Prosedur aproksimasi (pendekatan) yang umum dikenal adalah prosedur terbuka dan prosedur tertutup ^[1]. Prosedur Tertutup

Prosedur ini meliputi Prosedur Tabel, Prosedur Biseksi (Bagi-Dua) dan Prosedur Regula-Falsi. Prosedur tertutup memiliki ciri khas sebagai berikut: 1) Iterasi pencarian akar dilakukan pada interval $[x_1, x_2]$ tertentu, 2) Pada interval $[x_1, x_2]$ dapat dipastikan akan ditemukan satu akar yang dicari, 3) Hasil iterasi pencarian akar dipastikan konvergen, sehingga sering dikenal dengan nama prosedur konvergen.

1. Prosedur Terbuka

Prosedur ini meliputi Prosedur Newton-Raphson dan Prosedur Secant. Prosedur terbuka memiliki ciri khas sebagai berikut: 1) Membutuhkan masukan nilai awal, 2) Nilai x_n digunakan untuk menghitung nilai x_{n+1} , 3) Nilai x_{n+1} setelah iterasi berhenti dimungkinkan bersifat konvergen menuju bilangan riil tertentu atau divergen menuju 0 atau menuju $-\infty$ (bilangan riil negatif yang sangat besar) atau $+\infty$ (bilangan riil positif yang sangat besar).

Prosedur Biseksi (Bagi-Dua)

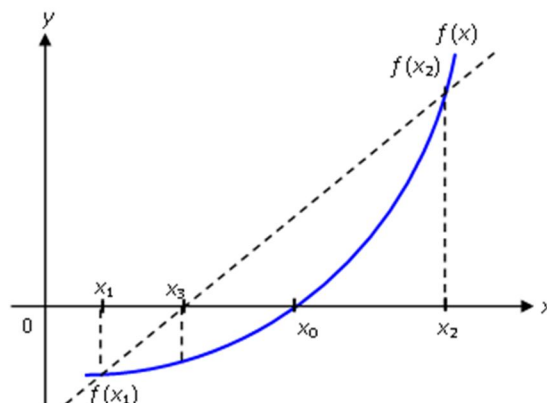


Gambar 2. Grafik Pencarian Akar dengan Prosedur Interval Bagi-Dua

Metoda ini sering digunakan untuk pencarian akar suatu fungsi riil non linear. Diumpamakan kita mengetahui bahwa suatu fungsi riil $f(x)$ memiliki satu akar pada interval $[a, b]$, atau satu akar terletak di antara titik $x = a$ dan $x = b$, maka $f(a)$ dan $f(b)$ memiliki tanda berlawanan, dengan asumsi bahwa bahwa grafik fungsi riil $f(x)$ merupakan fungsi kontinu, artinya garis grafik tidak terputus sepanjang interval $[a, b]$. Selanjutnya akan terlihat bahwa titik c merupakan tengah-tengah yang terletak pada interval $[a, b]$, atau dapat dinyatakan sebagai rata-rata a dan b , yaitu $c = \frac{1}{2}(a + b)$, yang selanjutnya menghasilkan nilai $f(c)$. Apabila $f(c)$ memiliki tanda yang sama dengan tanda $f(a)$, maka posisi akarnya pada interval $[c, b]$, atau kemungkinan lain posisi akarnya pada interval $[a, c]$.

Langkah berikutnya dilakukan pengurangan lebar interval dalam menjadi setengah dari lebar interval sebelumnya, demikian seterusnya diulang secara terus menerus menjadi $\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \dots$ hingga diperoleh akar yang kita cari dengan akurasi sesuai dengan yang diinginkan^[3].

Prosedur Regula-Falsi (False-Position)



Gambar 3. Grafik Pencarian Akar dengan Prosedur Regula-Falsi

Prosedur Regula-Falsi dikenal dengan prosedur perbaikan dari prosedur Biseksi (Bagi-Dua) yang dianggap kurang efisien. Salah satu kelemahan metoda Biseksi (Bagi-Dua) adalah ketika melakukan pembagian interval mulai dari x_{lower} hingga x_{upper} menjadi dua bagian yang sama besar, nilai $f(x_1)$ dan $f(x_2)$ tidak menjadi sesuatu yang penting. Sebagai contoh apakah $f(x_1)$ lebih dekat ke nol atau ke $f(x_n)$. Segitiga sebangun yang terbentuk dapat diperoleh taksiran perpotongan garis dan sumbu ^[4]

3. METODE PENELITIAN

Dalam penelitian ini, penulis menyusun langkah-langkah perhitungan iterasi beserta flow chart (diagram alir) untuk masing-masing prosedur yang akan diperbandingkan hasilnya, yaitu prosedur Interval Bagi-Dua dan Regula-Falsi, dengan formulasinya masing-masing.

Langkah-langkah Iterasi Numeri dengan Prosedur Interval Bagi-Dua (*Interval Bisection*) :

1. Tentukan nilai awal x_0 dan x_1 , selanjutnya hitung nilai $f(x_0)$ dan $f(x_1)$
2. Cek konvergensi nilai $f(x_0)$ dan $f(x_1)$, apabila:
 - a. Tanda $f(x_0) \neq$ tanda $f(x_1)$, maka nilai awal x_0 dan x_1 digunakan untuk iterasi selanjutnya (langkah 3).
 - b. Tanda $f(x_0) =$ tanda $f(x_1)$, maka tentukan nilai awal x_0 dan x_1 yang baru.
3. Hitunglah nilai x_t yang pertama dengan formula: $x_t = \frac{1}{2}(x_0 + x_1)$, kemudian tentukan keberadaan akar persamaan, di dalam sub interval mana berada, dengan cara:
 - a. Jika $f(x_0) < 0$, dan $f(x_1) > 0$, serta $f(x_t) < 0$, atau $f(x_0) > 0$, dan $f(x_1) < 0$, serta $f(x_t) > 0$, maka akar yang dicari berada pada sub interval pertama, kemudian tetapkan $x_2 = x_t$ dan lanjutkan pada iterasi kedua (langkah 4).
 - b. Jika $f(x_0) < 0$, dan $f(x_1) > 0$, serta $f(x_t) > 0$, atau $f(x_0) > 0$, dan $f(x_1) < 0$, serta $f(x_t) < 0$, maka akar persamaan berada pada sub interval kedua, kemudian tetapkan $x_3 = x_t$ dan lanjutkan pada iterasi kedua (langkah 4).
4. Hitunglah nilai x_t yang kedua dengan formula: $x_t = \frac{1}{2}(x_2 + x_3)$, kemudian tentukan keberadaan akar persamaan, di dalam sub interval mana berada, dengan cara:
 - a. Jika $f(x_2) < 0$, dan $f(x_3) > 0$, serta $f(x_t) < 0$, atau $f(x_2) > 0$, dan $f(x_3) < 0$, serta $f(x_t) > 0$, maka akar persamaan berada pada sub interval pertama, kemudian tetapkan $x_4 = x_t$ dan lanjutkan pada iterasi ketiga (langkah 5).
 - b. Jika $f(x_2) < 0$, dan $f(x_3) > 0$, serta $f(x_t) > 0$, atau $f(x_2) > 0$, dan $f(x_3) < 0$, serta $f(x_t) < 0$, maka akar persamaan berada pada sub interval kedua, kemudian tetapkan $x_5 = x_t$ dan lanjutkan pada iterasi ketiga (langkah 5).
5. Ulangilah langkah 4 untuk $x_t = \frac{1}{2}(x_n + x_{n+1})$,

Ketika taksiran baru sudah dianggap kecil (mendekati nol), maka perhitungan iterasi selesai, dan nilai x_t yang diperoleh merupakan hasil pencarian akar yang dimaksud. Apabila belum diperoleh taksiran baru yang nilainya cukup kecil (mendekati nol), maka perhitungan iterasi diulang kembali

sebagaimana pada langkah 4. Setelah taksiran baru sudah mendekati nol, atau dengan kata lain nilai $f(x_n) \cdot f(x_{n+1})$ mendekati 0, selanjutnya dilakukan perhitungan Mutlak Galat Relatif, dengan formula sebagai berikut .

$$\left| \sum x_{n+1} \right| = \frac{|x_{n+1} - x_n|}{|x_{n+1}|}$$

Dalam setiap langkah iterasi perhitungan pencarian akar, nilai Mutlak Galat Relatif ini diperbandingkan dengan besaran Toleransi atau nilai *Error*. Apabila besar nilai Mutlak Galat Relatif tersebut masih melebihi besar nilai Toleransi atau nilai *Error*, maka proses perhitungan akan dilanjutkan, sedemikian sehingga diperoleh nilai Mutlak Galat Relatif lebih kecil daripada nilai Toleransi *Error*.

Hasil perhitungan dari proses atau iterasi pencarian akar dapat dituangkan ke dalam tabel iterasi, sebagaimana ditunjukkan pada tabel 1.

Tabel 1. Formulasi Proses Iterasi dengan Prosedur Interval Bagi-Dua (*Interval Bisection*)

Iterasi	x_n	x_{n+1}	x_t	$f(x_n)$	$f(x_{n+1})$	$f(x_t)$	$\left \sum x_{n+1} \right $
0							
1							
..	..	..	..	..	..	..	
□	..	..	...	..			

Langkah-langkah Iterasi Numeri dengan Prosedur Regula-Falsi (*False-Position*)

1. Tentukan nilai awal x_0 dan x_1 , selanjutnya hitung nilai $f(x_0)$ dan $f(x_1)$
2. Cek konvergensi nilai $f(x_0)$ dan $f(x_1)$, apabila:
 - a. Tanda $f(x_0) \neq$ tanda $f(x_1)$, maka nilai awal x_0 dan x_1 dapat digunakan. untuk tahap iterasi selanjutnya (langkah 3).
 - b. Tanda $f(x_0) =$ tanda $f(x_1)$, maka tentukan nilai awal x_0 dan x_1 yang baru.
3. Hitunglan nilai x_2 dan seterusnya dengan rumus:

$$x_{n+1} = \frac{x_n \cdot f(x_{n-1}) - x_{n-1} \cdot f(x_n)}{(f(x_{n-1}) - f(x_n))}$$

4. Hasil perhitungan dari proses pencarian akar dapat dituangkan ke dalam tabel iterasi, sebagaimana ditunjukkan pada tabel 2.
5. Apabila diperoleh nilai x_{n+1} mendekati nilai x_n , maka proses iterasi selesai, dan x_{n+1} dinyatakan sebagai akar persamaan yang dicari, jika x_{n+1} belum mendekati nilai x_n , maka proses perhitungan kembali ke langkah 3. Penentuan nilai nilai x_{n+1} mendekati nilai x_n , ditentukan dengan membandingkannya dengan nilai Mutlak Galat Relatif, yang besarnya adalah sebagai berikut.

$$|\sum x_{n+1}| = \frac{|x_{n+1} - x_n|}{|x_{n+1}|}$$

Pada setiap tahapan proses pencarian akar (iterasi), jika nilai Mutlak Galat Relatif masih lebih besar dari nilai Toleransi *Error* maka proses atau iterasi pencarian akar dilanjutkan. Hasil perhitungan dari proses pencarian akar dapat dituangkan ke dalam tabel iterasi, sebagaimana ditunjukkan pada tabel 2.

Tabel 2. Formulasi Proses Iterasi dengan Prosedur Regula-Falsi

n	$x_{n+1} = \frac{x_n \cdot f(x_{n-1}) - x_{n-1} \cdot f(x_n)}{(f(x_{n-1}) - f(x_n))}$	$f(x_n)$	$ \sum x_{n+1} $
0	$x_0 = \dots$		
1	$x_1 = \dots$		
2			
...			

4. HASIL DAN PEMBAHASAN

Pada rangkaian penelitian ini, penulis memilih bahasa pemrograman Java untuk kedua prosedur yang diambil, baik untuk prosedur Prosedur Interval Bagi-Dua (*Interval Bisection*) dan prosedur Regula-Falsi (*False-Position*), pada fungsi polinomial dengan persamaan:
 $f(x) = 2x^2 + 3x^2 - 6x - 16$

Prosedur *Interval Bagi-Dua*

Hasil Keluaran Program Java pada Prosedur **Interval Bagi-Dua**

run:

Pencarian akar persamaan suku banyak: $f(x)=2X^3 + 3X^2 - 6X - 16$, dengan prosedur Interval Bagi-Dua. Masukan Awal: $x_0 = 1$, $x_1 = 4$; toleransi = 0,005

Tabel 3. Hasil Proses Iterasi dengan Prosedur Interval Bagi-Dua

No	x_n	x_{n+1}	x_t	$f(x_n)$	$f(x_{n+1})$	$f(x_t)$	$ \sum x_{n+1} $
0	1	4	2,5	-17,00000	136,00000	19,00000	0,60000
1	1	2,5	1,75	-17,00000	19,00000	-6,59375	0,42857
2	1,75	2,5	2,125	-6,59375	19,00000	3,98828	0,17647
3	1,75	2,125	1,9375	-6,59375	3,98828	-1,81689	0,09677
4	1,9375	2,125	2,03125	-1,81689	3,98828	0,95221	0,04615
5	1,9375	2,03125	1,98438	-1,81689	0,95221	-0,46510	0,02362
6	1,98437	2,03125	2,00781	-0,46510	0,95221	0,23529	0,01167
7	1,98437	2,00781	1,99609	-0,46510	0,23529	-0,11696	0,00587
8	1,99609	2,00781	2,00195	-0,11696	0,23529	0,05865	0,00293

Nilai Akar Persamaan yang diperoleh : 2,00195

Total Waktu Iterasi : 0 detik

Dari tabel 3 menunjukkan bahwa proses pencarian akar dengan menggunakan prosedur Interval Bagi-Dua, jumlah tahapan iterasi atau langkah penyelesaian untuk pencarian akar persamaan adalah sebanyak 9 langkah.

Prosedur Regula-Falsi

Program Java untuk Prosedur Regula-Falsi

Output Program Java untuk Prosedur Interval Regula-Falsi

Pencarian akar persamaan suku banyak: $f(x)=2X^3 + 3X^2 - 6X - 16$, dengan Prosedur Regula-Falsi. Masukan Awal: $x_0 = 1$, $x_1 = 4$; toleransi = 0,005

Tabel 4. Hasil Proses Iterasi dengan Prosedur Regula-Falsi

n	$x_{n+1} = \frac{x_n \cdot f(x_{n-1}) - x_{n-1} \cdot f(x_n)}{(f(x_{n-1}) - f(x_n))}$	$f(x_n)$	$ \sum x_{n+1} $
0	1	-17,00000	
1	4	136,00000	0,75000
2	1,33333	-13,92593	2,00000
3	1,58103	-10,08319	0,15667
4	2,23097	7,75391	0,29133
5	1,94844	-1,50734	0,14501
6	1,99442	-0,16695	0,02306
7	2,00015	0,00441	0,00286

Nilai Akar Persamaan yang diperoleh : 2,00015

Total Waktu Iterasi : 0 detik

Dari tabel 4 menunjukkan bahwa proses pencarian akar dengan menggunakan prosedur Interval Regula-Falsi, jumlah tahapan iterasi atau langkah penyelesaian untuk pencarian akar persamaan adalah sebanyak 8 langkah.

5. KESIMPULAN

Kesimpulan yang diperoleh dari paparan hasil penelitian di atas, yang mencoba membandingkan hasil iterasi dengan prosedur Interval Bagi-Dua dan prosedur Regula-Falsi, ternyata diperoleh beberapa hasil sebagai berikut :

1. Laju proses iterasi untuk mencari nilai akar fungsi polinomial berderajat 3 pada prosedur Interval Bagi-Dua adalah 9 langkah, sedangkan pada prosedur Regula-Falsi ada 8 langkah.
2. Nilai akar pada studi kasus untuk fungsi dengan menggunakan prosedur Interval Bagi-Dua adalah 2,00195, dengan galat sebesar 0,00293, sedangkan dengan menggunakan prosedur Regula-Falsi 2,00015, dengan galat sebesar 0,00283,
3. Iterasi dengan prosedur Regula-Falsi menunjukkan akselerasi yang lebih baik dibandingkan iterasi dengan prosedur Interval Bagi-Dua, dengan galat relative yang lebih baik pula.

UCAPAN TERIMA KASIH

Terimakasih yang sebesar-besarnya saya sampaikan kepada dosen pembimbing, para staf administrasi STMIK AKI, BAAK dan BAAU STMIK AKI, serta para dosen STMIK AKI, khususnya kepada ketua STMIK AKI, serta kepada seluruh pihak yang telah membantu dalam kegiatan penelitian yang dilakukan.

DAFTAR PUSTAKA

- [1] Chapra, Steven C dan Raymond P. Canale (2010). *Numerical Procedures for Engineers*. McGraw-Hill Companies, Inc.
- [2] Endang Sunandar, Indrianto, (2020). *Perbandingan Prosedur Newton-Raphson & Prosedur Secant Untuk Mencari Akar Persamaan Dalam Sistem Persamaan Non-Linier*, PETIR: Jurnal Pengkajian dan Penerapan Teknik Informatika, Vol.13, No.1, hal.72-79.
- [3] Chasnov, Jeffrey R., (2012). *Introduction to Numerical Procedures Lecture notes*, The Hong Kong University of Science and Technology Department of Mathematics.
- [4] Wulan ER, Sukarti SM, Zulkarnaen D. (2017). *Perbandingan Tingkat Kecepatan Konvergensi dari Prosedur Newton Raphson dan Prosedur Secant Setelah Mengaplikasikan Prosedur Aiken's dalam Perhitungan Akar Pangkat Tiga*. Jurnal Matematika Integratif ISSN.1412:6184.
- [5] Sahid, (2003). *Analisis dan Implementasi Prosedur Newton Raphson*, Jurusan Matematika Fakultas FMIPA, Universitas Negeri Yogyakarta.
- [6] Elis Ratna Wulan dkk, (2016). *Perbandingan Tingkat Kecepatan Konvergensi dari Prosedur Newton Raphson dan Prosedur Secant Setelah Mengaplikasikan Prosedur Aiken's dalam Perhitungan Akar Pangkat Tiga*. Jurnal Matematika Integratif, Vol.12, No.1, pp 35–42.
- [7] E. Sunandar (2019). *Perbandingan Prosedur Selection Sort dan Insertion Sort Dalam Pengurutan Data Menggunakan Bahasa Program Java*. petir, vol. 12, no. 2, pp. 172 - 178.