**MENCARI AKAR PERSAMAAN NON-LINEAR**

Disusun oleh : Kelompok 6 - Kelas E - Informatika 2024

* Revanska Muhammad Athalllah (24060124140129)
* Reynaldi Bertinus Hutagaol (24060124140157)
* Ruth Septriana Sipangkar (24060124120024)
* Silvani Salsabilla (24060124130066)
* Yustinus Hendi Setyawan (24060124130114)

1. **Pendahuluan Mencari Akar Persamaan Non-Linear**

Akar sebuah persamaan non-linier merupakan nilai x yang menyebabkan nilai f(x) sama dengan nol. Dalam hal ini dapat disimpulkan bahwa akar-akar penyelesaian persamaan non-linier merupakan titik potong antara kurva f(x) dengan sumbu x.

Contoh sederhana dari penentuan akar persamaan non-linier adalah penentuan akar persamaan kuadratik. Secara analitik penentuan akar persamaan kuadratik dapat dilakukan menggunakan persamaan berikut.

x12 =

Untuk masalah yang lebih rumit, penyelesaian analitik sudah tidak mungkin dilakukan. Metode numerik dapat digunakan untuk menyelesaikan masalah yang lebih kompleks. Untuk mengetahui apakah suatu persamaan non-linier memiliki akar-akar penyelesaian atau tidak, diperlukan analisa menggunakan Teorema berikut:

*Suatu range x=[a,b] mempunyai akar bila f(a) dan f(b) berlawanan tanda atau memenuhi f(a).f(b)<0.*

1. **Metode Mencari Akar Persamaan Non-Linear**
2. **Metode Biseksi**

Metode ini bekerja berdasarkan **teorema Bolzano**, yang menyatakan bahwa jika suatu fungsi kontinu dalam interval [a,b][a, b] memiliki tanda yang berlawanan di ujung interval tersebut, maka terdapat setidaknya satu akar di dalamnya.

Metode biseksi membagi interval menjadi dua bagian secara berulang-ulang hingga diperoleh nilai akar yang mendekati hasil yang diinginkan dengan tingkat ketelitian tertentu.

Konsep dasar metode biseksi mirip dengan metode tabel, di mana area pencarian akar dibagi menjadi beberapa bagian. Namun, dalam metode biseksi, interval selalu dibagi menjadi dua bagian yang sama. Dari dua bagian tersebut, bagian yang tidak mengandung akar dibuang, dan bagian yang mengandung akar tetap digunakan. Proses ini dilakukan secara berulang-ulang hingga diperoleh akar dengan tingkat ketelitian yang telah ditentukan.

Metode biseksi adalah metode iteratif yang membagi interval menjadi dua bagian dan memilih subinterval yang mengandung akar. Interval yang tidak mengandung akar dibuang, dan proses ini dilakukan secara berulang hingga diperoleh akar yang mendekati nilai sebenarnya.

**Algoritma**

1. Tentukan fungsi f(x) yang akan dicari akarnya.
2. Tentukan nilai batas bawah (a) dan batas atas (b) dengan syarat f(a)⋅f(b)<0, yang berarti fungsi memiliki tanda berlawanan di kedua batas tersebut.
3. Tentukan toleransi kesalahan ϵ\epsilon dan jumlah iterasi maksimum N.
4. Hitung nilai tengah:

x =

1. Evaluasi nilai fungsi di titik tengah, yaitu f(x).
2. Perbarui batas interval:
   * Jika f(a)⋅f(x) < 0 maka akar berada di interval [x,a], sehingga diperbarui b = x
   * Jika f(b)⋅f(x )< 0 maka akar berada di interval [x,b] , sehingga diperbarui a = x

7. Periksa kondisi penghentian:

* Jika ∣f(x)∣< ϵ atau jumlah iterasi melebihi N, proses dihentikan dan akar diambil sebagai nilai x.
* Jika tidak, ulangi dari langkah 4.

**Contoh Penerapan**

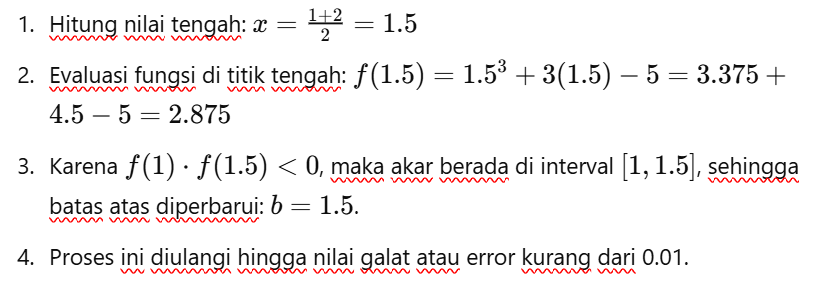
Misalkan kita ingin mencari akar dari persamaan berikut:

f(x) = x3 + 3x - 5

dengan batas awal :

* Batas bawah a=1
* Batas atas b=2
* Toleransi kesalahan ϵ=0.01

Langkah-langkahnya:

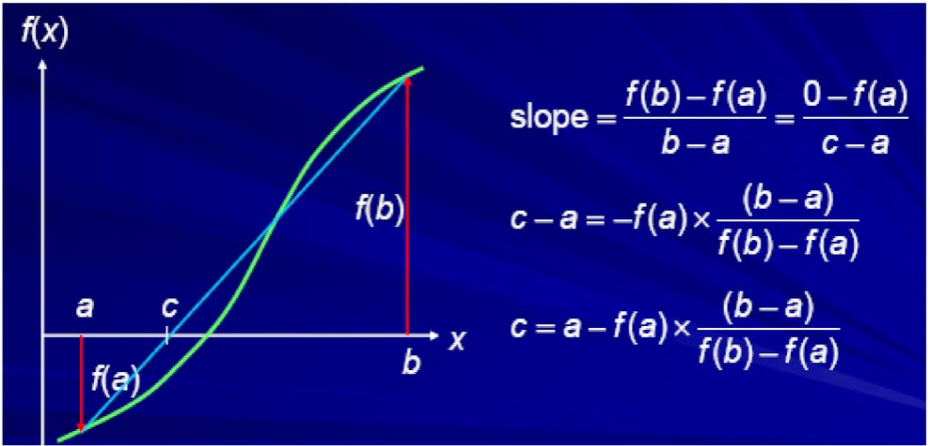
.

Setelah iterasi sebanyak 8 kali, diperoleh nilai akar yang mendekati dengan kesalahan di bawah 0.01.

1. **Metode Regula Falsi**

Metode regula falsi adalah sebuah metode yang digunakan untuk pencarian akar persamaan dengan memanfaatkan kemiringan dan selisih tinggi dari 2 titik batas range.

Metode ini memiliki kecepatan konvergensi yang lebih cepat dari metode biseksi.



gradien garis AB = gradien BX

= 🡪 x = b -

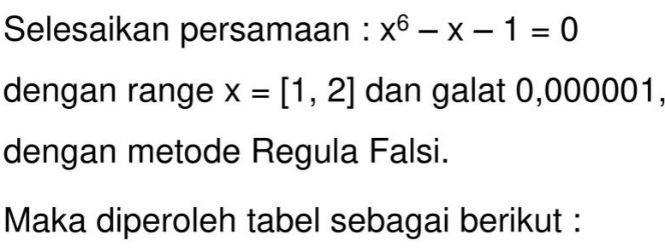
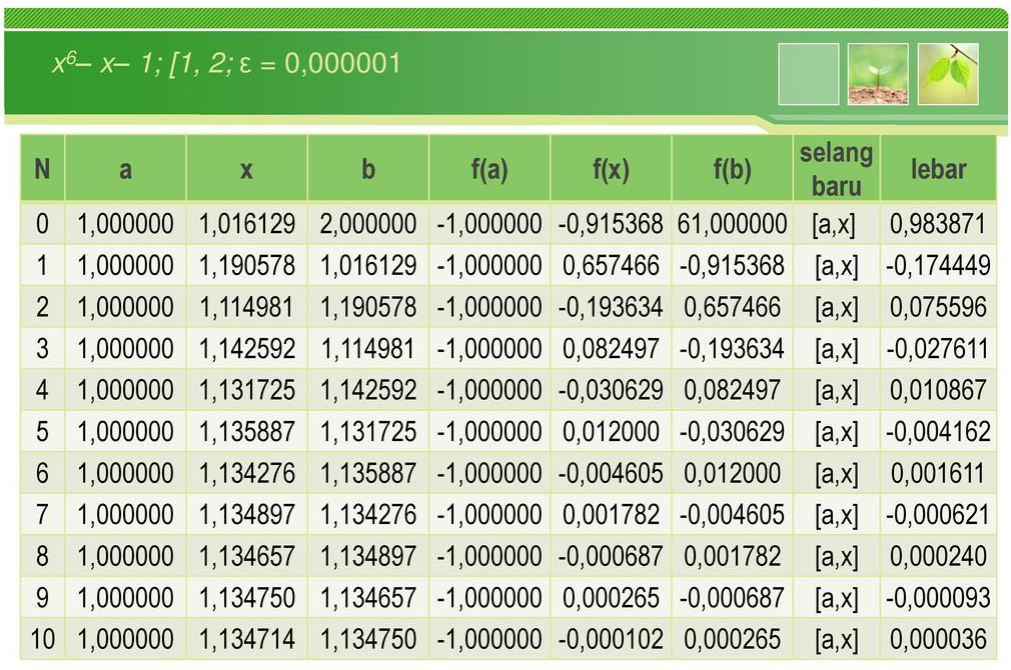
**Algoritma**

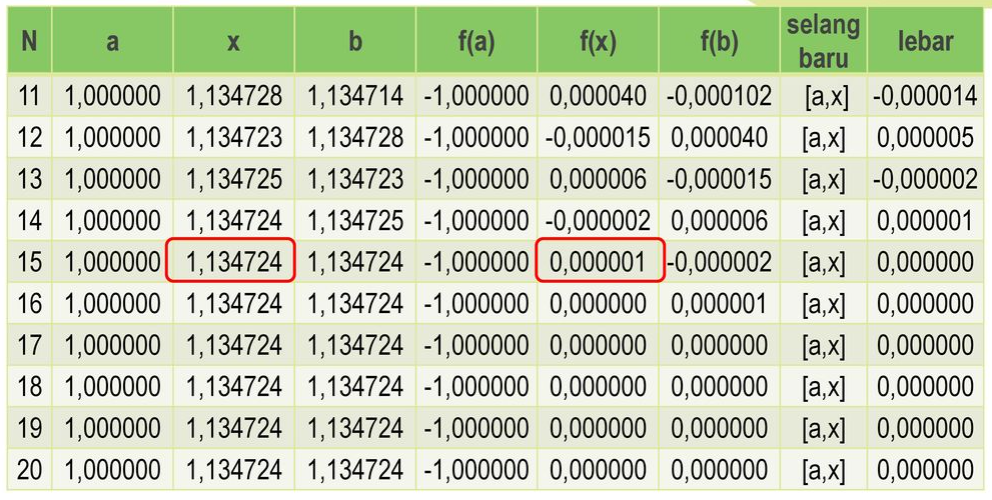
1. Definisikan f(x) yang akan dicari akarnya
2. tentukan nilai batas bawah (a) dan batas atas (b)
3. tentukan toleransi nilai dan iterasi maksimum N
4. Hitung F(a) = f(a) dan F(b) = f(b)
5. Untuk iterasi I = 1 s/d N atau > 0

* x = b -
* Hitung F(x) = f(x)
* Hitung error = |F(x)|
* Jika Fx.Fa < 0 maka b = x dan Fb = Fx,jika tidak a = x dan Fa = Fx

1. Akar Persamaan adalah x

**Contoh**



Dari tabel diatas diperoleh f(x) = 0,000001 adalah nilai yang paling mendekati 0 pada ɛ = 0,000001,sehingga hampiran akarnya adalah 1,134724

1. **Metode Newton-Rapshon**

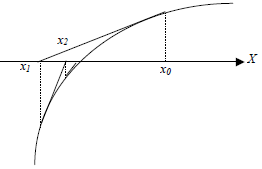
**Definisi**

Merupakan metode penyelesaian persamaan non-linier dengan pendekatan satu titik awal dan mendekatinya dengan memperhatikan slope atau gradien, dengan kata lain menggunakan informasi dari fungsi dan turunan titik untuk mendapatkan tebakan yang baik.

**Asumsi :**

* f(x) kontinu dan turunan pertamanya diketahui
* Telah diberikan tebakan awal x0 sedemikian sehingga f(x0) = 0

**Ilustrasi**



**Algoritma**

1. Definisikan f(x) dan f’(x)
2. Tentukan nilai toleransi e dan iterasi maksimum (N)
3. Tentukan tebakan awal x0
4. Hitung f(x0) dan f’(x0)
5. Untuk iterasi i=1 s/d N atau |f(x)| , hitung x menggunakan persamaan



1. Akar Persamaan merupakan nilai xi terakhir yang diperoleh.

**Contoh Pengerjaan**

Carilah akar persamaan dari *f*(*x0*)= x−*e-x*=0

Solusi:

1. Definisikan f(x) dan f'(x)



1. Tentukan nilai toleransi e dan iterasi maksimum (N)

Disini kami memilih nilai toleransi =10-6 N = 100

1. Tentukan tebakan awal x0

Disini kami memilih nilai awal (x0) = 0

1. Proses iterasi

**Iterasi Satu:** Menghitung f(x0) dan f’(x0)

*f*(*x0*​)=0−*e0*=−1

*f*′(*x0*​)=1+*e0*=2

Masukkan ke persamaan Newton

x1=x0-f(x0)f′(x0)=0-12=0.5

**Iterasi Dua:** Menghitung f(*x1*​) dan f’(*x1*​)

*x1*​=0.5

*f*(*x1*​)=0.5−*e-0,5* ≈ 0.5−0.6065=−0.1065

*f*′(*x1*​)=1+*e-0,5* ≈ 1+0.6065=1.6065

Masukkan ke persamaan Newton

x2=x1−f(x1)f′(x1)=0.5−-0.10651.6065 ≈ 0.5663

**Iterasi Tiga:** Menghitung f(x2) dan f’(x2)

x2=0.5663

*f*(*x2*)=0.5663−*e-*0.5663≈ 0.5663−0.5675 = -0.0012

f′(x2)=1+e−0.5663 ≈ 1+0.5675=1.5675

Masukkan ke persamaan Newton

x3=x2− f(x2)f′(x2)=0.5663 --0.00121.5675≈ 0.5671

Hitung nilai

∣*x3*​−*x2*​∣=∣0.5671−0.5663∣=0.0008

Karena belum memenuhi toleransi maka iterasi dilanjutkan.

**Iterasi Empat :** Menghitung f(x3) dan f’(x3)

*x3*=0.5671

*f*(*x3*​)=0.5671−*e*−0.5671≈ 0.5671−0.5671=0

f′(x3)=1+e−0.5671 ≈ 1+0.5671=1.5671

Masukkan ke persamaan Newton

=*x3*−*f*(*x3*​)*f*′(*x*33)​=0.5671−0​=0.5671

Hitung nilai

*x4*​−*x3*​∣=∣0.5671−0.5671∣=0

Memenuhi toleransi

Maka ditemukan akar dari *f*(*x0*)= x−*e-x*=0 adalah 0.567143 yang ditemukan pada iterasi keempat.

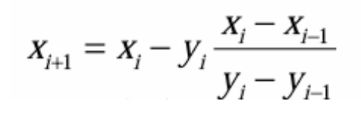
1. **Metode Secant**

**Definisi**

Metode Secant merupakan salah satu algoritma numerik yang efisien dalam mencari akar suatu fungsi non-linear. Prinsip kerjanya didasarkan pada aproksimasi garis singgung pada kurva fungsi di dua titik yang berdekatan. Titik potong garis singgung ini dengan sumbu-x kemudian dijadikan sebagai aproksimasi akar berikutnya. Proses iterasi ini berulang hingga diperoleh tingkat akurasi yang diinginkan.

**Algoritma**

1. Definisikan fungsi F(x)
2. Definisikan torelansi error (e) dan iterasi maksimum (n)
3. Masukkan dua nilai pendekatan awal yang di antaranya terdapat akar yaitu x0 dan x1, sebaiknya gunakan metode tabel atau grafis untuk menjamin titik pendakatannya adalah titik pendekatan yang konvergensinya pada akar persamaan yang diharapkan.
4. Hitung F(x0) dan F(x1) sebagai y0 dan y1
5. Untuk iterasi I = 1 s/d n atau |F(xi)|



1. Hitung yi+1 = F(Xi+1)
2. Akar persamaan adalah nilai x yang terakhir.

**Contoh Soal**

