UNIVERSITAS SULTAN AGENG TIRTAYASA

UNTIRTA Jawara
Jujur Adil Wibawa Amanah Religius Akuntabel

Komputasi Statistika

#10 Meeting

Root Finding : Fixed Poin & Newton Raphson

Ferdian Bangkit Wijaya, S.Stat., M.Si
NIP. 199005202024061001

Root Finding

- **Definisi:** "Pencarian Akar" (*Root Finding*) adalah proses mencari nilai x yang membuat sebuah fungsi $f(x)$ bernilai **nol**. Nilai x ini disebut **"akar"** (root) dari fungsi tersebut.
- **Contoh Sederhana:** Kita punya fungsi $f(x) = x - 5$.
  - Mencari akarnya berarti mencari x sehingga $f(x) = 0$.
  - $x - 5 = 0$
  - Jawabannya (akarnya) jelas adalah $x = 5$.

Ini mudah karena kita bisa menyelesaikannya secara aljabar. Tapi bagaimana jika fungsinya $f(x) = \cos(x) - x$? Tidak ada cara aljabar untuk menyelesaikannya. Di sinilah kita butuh metode numerik, seperti *Fixed-Point* dan *Newton-Raphson*, yang menggunakan pendekatan iteratif (langkah-demi-langkah) untuk *menghampiri* jawaban.

Root Finding

Metode Fixed Poin

Metode ini mengubah masalah $f(x) = 0$ menjadi masalah "titik tetap" (fixed point) $g(x) = x$.

1. **Transformasi:** Kita ubah $f(x) = 0$ secara aljabar ke bentuk $g(x) = x$.
   - Misal: $f(x) = \cos(x) - x = 0$. Ini bisa langsung diubah menjadi $g(x) = \cos(x)$. Jadi, kita mencari x di mana $x = \cos(x)$.

2. **Iterasi:** Kita mulai dengan tebakan awal (x_0), lalu terus-menerus memasukkan hasilnya kembali ke fungsi $g(x)$.
   - $x_1 = g(x_0)$
   - $x_2 = g(x_1)$
   - $x_3 = g(x_2)$
   - ... dan seterusnya, sampai $x_{n+1} \approx x_n$.

Root Finding

Metode Fixed Poin

3. **Konvergensi:** Metode ini hanya berhasil (konvergen) jika nilai absolut turunan $g(x)$, yaitu $|g'(x)|$, **lebih kecil dari 1** di sekitar akar. Jika $|g'(x)| > 1$, iterasinya akan menjauh (divergen).

Contoh 1: Mencari Akar dari $f(x) = \cos(x) - x$

Kita ingin mencari x di mana $\cos(x) = x$.

- $f(x) = \cos(x) - x$
- Kita ubah menjadi $g(x) = \cos(x)$.
- Kita cek turunannya: $g'(x) = -\sin(x)$. Di sekitar akar (sekitar 0.7), $|-\sin(0.7)| \approx 0.64$, yang mana < 1. Jadi, ini akan konvergen.
- Kita mulai dengan tebakan awal, misal $x_0 = 0.5$.

Root Finding

Metode Fixed Poin (Contoh 1)

UNIVERSITAS SULTAN AGENG TIRTAYASA

UNTIRTA Jawara
Jujur Adil Wibawa Amanah Religius Akuntabel

Mencari Akar dari $f(x) = \cos(x) - x$

$g(x) = \cos(x)$ | Tebakan Awal $x_0 = 0.5$

Iterasi 1:

- $x_1 = g(x_0) = \cos(0.5) \approx 0.877583$
- Error = $|x_1 - x_0| = |0.877583 - 0.5| \approx \mathbf{0.377583}$

Iterasi 2:

- $x_2 = g(x_1) = \cos(0.877583) \approx 0.639012$
- Error = $|x_2 - x_1| = |0.639012 - 0.877583| \approx \mathbf{0.238571}$

Iterasi 3:

- $x_3 = g(x_2) = \cos(0.639012) \approx 0.802685$
- Error = $|x_3 - x_2| = |0.802685 - 0.639012| \approx \mathbf{0.163673}$

Root Finding

Metode Fixed Poin (Contoh 1)

```
# 1. Definisikan fungsi g(x) kita
g <- function(x) {
  return(cos(x))
}

# 2. Inisialisasi
x_old <- 0.5      # Tebakan awal (x0)
tolerance <- 1e-7 # Batas toleransi error
max_iter <- 20    # Batas iterasi maksimum
iter <- 0
error <- 1

# Siapkan tabel untuk hasil
results <- data.frame(Iterasi = integer(),
                      xn = double(),
                      Error = double())
```

```
# 3. Proses Iterasi
while (error > tolerance && iter < max_iter) {
  iter <- iter + 1
  x_new <- g(x_old)          # Hitung x_n+1 = g(x_n)
  error <- abs(x_new - x_old) # Hitung error

  # Simpan hasil
  results[iter, ] <- c(iter, x_new, error)

  # Update x_old untuk iterasi berikutnya
  x_old <- x_new
}

# Tampilkan hasil
print(results, digits = 8)
cat("\nAkarnya adalah:", x_new, "\n")
```

Root Finding

Metode Fixed Poin (Contoh 1)

```
> print(results, digits = 8)
   Iterasi         xn         Error
1        1 0.87758256 0.37758256189
2        2 0.63901249 0.23857006773
3        3 0.80268510 0.16367260652
4        4 0.69477803 0.10790707389
5        5 0.76819583 0.07341780449
6        6 0.71916545 0.04903038534
7        7 0.75235576 0.03319031348
8        8 0.73008106 0.02227469628
9        9 0.74512034 0.01503927821
10      10 0.73500631 0.01011403234
11      11 0.74182652 0.00682021363
12      12 0.73723573 0.00459079720
13      13 0.74032965 0.00309392644
14      14 0.73824624 0.00208341355
15      15 0.73964996 0.00140372444
16      16 0.73870454 0.00094542341
17      17 0.73934145 0.00063691292
18      18 0.73891245 0.00042900295
19      19 0.73920144 0.00028899480
20      20 0.73900678 0.00019466435
> cat("\nAkarnya adalah:", x_new, "\n")

Akarnya adalah: 0.7390068
```

Root Finding

Metode Fixed Poin (Contoh 2)

Mencari Akar dari $f(x) = x^2 - x - 2 = 0$

Bentuk $g(x) = x$ bisa dibuat macam-macam:

1. **Bentuk A:** $x^2 = x + 2 \implies x = \sqrt{x+2}$. Mari kita gunakan $g(x) = \sqrt{x+2}$.
2. **Bentuk B:** $x(x-1) = 2 \implies x = 2/(x-1)$.
3. **Bentuk C:** $x^2 - 2 = x \implies x = x^2 - 2$.

Mari kita coba **Bentuk A:** $g(x) = \sqrt{x+2}$.

- Turunannya $g'(x) = 1/(2\sqrt{x+2})$.
- Di dekat $x = 2$, $g'(2) = 1/(2\sqrt{4}) = 1/4 = 0.25$. Ini < 1, jadi pasti konvergen.
- Mari kita mulai dengan tebakan $x_0 = 1.5$.

Root Finding

Metode Fixed Poin (Contoh 2)

Iterasi 1:

- $x_1 = g(x_0) = \sqrt{1.5 + 2} = \sqrt{3.5} \approx 1.870829$
- Error = $|x_1 - x_0| = |1.870829 - 1.5| \approx \mathbf{0.370829}$

Iterasi 2:

- $x_2 = g(x_1) = \sqrt{1.870829 + 2} = \sqrt{3.870829} \approx 1.967468$
- Error = $|x_2 - x_1| = |1.967468 - 1.870829| \approx \mathbf{0.096639}$

Iterasi 3:

- $x_3 = g(x_2) = \sqrt{1.967468 + 2} = \sqrt{3.967468} \approx 1.991851$
- Error = $|x_3 - x_2| = |1.991851 - 1.967468| \approx \mathbf{0.024383}$

Root Finding

Metode Fixed Poin (Contoh 2)

```
# 1. Definisikan g(x) = sqrt(x + 2)
g <- function(x) {
  return(sqrt(x + 2))
}

# 2. Inisialisasi
x_old <- 1.5      # Tebakan awal (x0)
tolerance <- 1e-7  # Batas toleransi error
max_iter <- 20     # Batas iterasi maksimum
iter <- 0
error <- 1

# Siapkan tabel untuk hasil
results <- data.frame(Iterasi = integer(),
                      xn = double(),
                      Error = double())

# 3. Proses Iterasi
while (error > tolerance && iter < max_iter) {
  iter <- iter + 1
  x_new <- g(x_old)
  error <- abs(x_new - x_old)

  results[iter, ] <- c(iter, x_new, error)
  x_old <- x_new
}

# Tampilkan hasil
print(results, digits = 8)
cat("\nAkarnya adalah:", x_new, "\n")
```

Root Finding

Metode Fixed Poin (Contoh 2)


```
> print(results, digits = 8)
   Iterasi        xn         Error
1        1 1.8708287 3.7082869e-01
2        2 1.9674422 9.6613477e-02
3        3 1.9918439 2.4401742e-02
4        4 1.9979599 6.1160254e-03
5        5 1.9994899 1.5299818e-03
6        6 1.9998725 3.8255642e-04
7        7 1.9999681 9.5642918e-05
8        8 1.9999920 2.3910968e-05
9        9 1.9999980 5.9777568e-06
10      10 1.9999995 1.4944401e-06
11      11 1.9999999 3.7361009e-07
12      12 2.0000000 9.3402526e-08
> cat("\nAkarnya adalah:", x_new, "\n")

Akarnya adalah: 2
```

Apakah akarnya hanya 2 ?

cek akar NEGATIF nya juga

Root Finding

Metode Fixed Poin (Contoh 2)

Kita harus mengubah $f(x) = 0$ menjadi bentuk $g(x) = x$ yang lain, yang "menarik" ke arah -1.

Ingat, $f(x) = x^2 - x - 2 = 0$. Bentuk lain yang bisa kita buat: $x^2 - x = 2\, x(x-1) = 2\, x = \frac{2}{x-1}$

Mari kita gunakan formulasi baru ini: $g(x) = \frac{2}{x-1}$.

Sekarang kita coba lagi dengan tebakan awal dekat -1.

- **Tebakan Awal (x_0):** $x_0 = -0.5$
- **Formulasi Baru $g(x)$:** $g(x) = 2/(x-1)$

Root Finding

Metode Fixed Poin (Contoh 2)

$g(x) = 2/(x-1)$ | **Tebakan Awal** $x_0 = -0.5$

Validasi (Cek Turunan):

- $g(x) = 2(x-1)^{-1}$
- $g'(x) = -2(x-1)^{-2} = \frac{-2}{(x-1)^2}$
- Kita cek di dekat akar target $x = -1$:
- $|g'(-1)| = |\frac{-2}{(-1-1)^2}| = |\frac{-2}{(-2)^2}| = |\frac{-2}{4}| = 0.5$

Karena $0.5 < 1$, metode ini **pasti konvergen**

Iterasi 1:

- $x_1 = g(x_0) = g(-0.5) = \frac{2}{-0.5-1} = \frac{2}{-1.5} \approx -1.333333$
- **Error** = $|x_1 - x_0| = |-1.333333 - (-0.5)| \approx \mathbf{0.833333}$

Iterasi 2:

- $x_2 = g(x_1) = g(-1.333333) = \frac{2}{-1.333333-1} = \frac{2}{-2.333333} \approx -0.857143$
- **Error** = $|x_2 - x_1| = |-0.857143 - (-1.333333)| \approx \mathbf{0.476190}$

Iterasi 3:

- $x_3 = g(x_2) = g(-0.857143) = \frac{2}{-0.857143-1} = \frac{2}{-1.857143} \approx -1.076923$
- **Error** = $|x_3 - x_2| = |-1.076923 - (-0.857143)| \approx \mathbf{0.219780}$

Root Finding

Metode Fixed Poin (Contoh 2)

Iterasi 4:

- $x_4 = g(x_3) = g(-1.076923) = \frac{2}{-1.076923-1} = \frac{2}{-2.076923} \approx -0.963043$
- Error = $|x_4 - x_3| = |-0.963043 - (-1.076923)| \approx \mathbf{0.113880}$

Iterasi 5:

- $x_5 = g(x_4) = g(-0.963043) = \frac{2}{-0.963043-1} = \frac{2}{-1.963043} \approx -1.018824$
- Error = $|x_5 - x_4| = |-1.018824 - (-0.963043)| \approx \mathbf{0.055781}$

...dan seterusnya. Kita bisa lihat angkanya "melompat-lompat" di sekitar -1 (disebut oscillating convergence), tapi errornya terus mengecil dan pasti akan menuju -1.

Root Finding

Metode Fixed Poin (Contoh 2)

```
# 1. Definisikan g(x) = 2 / (x - 1)
g <- function(x) {
  return(2 / (x - 1))
}

# 2. Inisialisasi
x_old <- -0.5      # Tebakan awal (x0)
tolerance <- 1e-7  # Batas toleransi error
max_iter <- 30     # Batas iterasi maksimum
iter <- 0
error <- 1

# Siapkan tabel untuk hasil
results <- data.frame(Iterasi = integer(),
                      xn = double(),
                      Error = double())
```

```
# 3. Proses Iterasi
while (error > tolerance && iter < max_iter) {
  iter <- iter + 1
  x_new <- g(x_old)
  error <- abs(x_new - x_old)

  results[iter, ] <- c(iter, x_new, error)
  x_old <- x_new
}

# Tampilkan hasil
print(results, digits = 8)
cat("\nAkarnya adalah:", x_new, "\n")
```

Root Finding

Metode Fixed Poin (Contoh 2)

```
> print(results, digits = 8)
   Iterasi          xn         Error
1        1 -1.33333333 8.3333333e-01
2        2 -0.85714286 4.7619048e-01
3        3 -1.07692308 2.1978022e-01
4        4 -0.96296296 1.1396011e-01
5        5 -1.01886792 5.5904962e-02
6        6 -0.99065421 2.8213719e-02
7        7 -1.00469484 1.4040630e-02
8        8 -0.99765808 7.0367561e-03
9        9 -1.00117233 3.5142533e-03
10      10 -0.99941418 1.7581560e-03
11      11 -1.00029300 8.7882044e-04
12      12 -0.99985352 4.3947459e-04
13      13 -1.00007324 2.1972120e-04
14      14 -0.99996338 1.0986462e-04
15      15 -1.00001831 5.4931305e-05
16      16 -0.99999084 2.7465904e-05
17      17 -1.00000458 1.3732889e-05
18      18 -0.99999771 6.8664603e-06
19      19 -1.00000114 3.4332262e-06
20      20 -0.99999943 1.7166141e-06
21      21 -1.00000029 8.5830680e-07
22      22 -0.99999986 4.2915346e-07
23      23 -1.00000007 2.1457672e-07
24      24 -0.99999996 1.0728836e-07
25      25 -1.00000002 5.3644180e-08
> cat("\nAkarnya adalah:", x_new, "\n")

Akarnya adalah: -1
```

Root Finding

Metode Newton Raphson

Metode Newton-Raphson (atau disebut juga Metode Newton) adalah salah satu metode pencarian akar yang paling cepat dan populer.

1. **Ide Dasar:** Metode ini menggunakan **garis singgung (turunan)** dari kurva fungsi untuk memprediksi di mana kurva itu akan memotong sumbu-x (akar).
2. **Visualisasi:**
   - Mulai dari tebakan x_0.
   - Hitung $f(x_0)$.
   - Buat garis singgung pada kurva di titik $(x_0, f(x_0))$.
   - Tebakan berikutnya, x_1, adalah titik di mana garis singgung itu memotong sumbu-x (sumbu nol).
   - Ulangi proses dari x_1 untuk mendapatkan x_2, dan seterusnya.

Root Finding

Metode Newton Raphson

3. **Rumus Iterasi:** Rumus ini didapat dari persamaan garis singgung: $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
   - $f(x_n)$ adalah nilai fungsi di x_n.
   - $f'(x_n)$ adalah nilai **turunan pertama** fungsi di x_n.
4. **Kelemahan:** Metode ini butuh turunan $f'(x)$, yang terkadang sulit dihitung. Metode ini juga bisa gagal (divergen) jika $f'(x_n)$ sangat dekat dengan nol (garis singgungnya datar).

Contoh 1: Mencari Akar dari $f(x) = \cos(x) - x$

- Fungsi: $f(x) = \cos(x) - x$
- Turunan: $f'(x) = -\sin(x) - 1$
- Rumus Iterasi: $x_{n+1} = x_n - \frac{\cos(x_n) - x_n}{-\sin(x_n) - 1}$
- Kita mulai dengan tebakan awal, misal $x_0 = 0.5$.

Root Finding

Metode Newton Raphson (Contoh 1)

$f(x) = \cos(x) - x$ | Tebakan Awal $x_0 = 0.5$

Iterasi: $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$

Iterasi 1:

- $f(0.5) \approx 0.37758$
- $f'(0.5) \approx -1.47942$
- $x_1 = 0.5 - (0.37758/-1.47942) \approx 0.5 - (-0.25522) \approx 0.75522$
- Error = $|x_1 - x_0| = |0.75522 - 0.5| \approx \mathbf{0.25522}$

Iterasi 2:

- $f(0.75522) \approx -0.02711$
- $f'(0.75522) \approx -1.68533$
- $x_2 = 0.75522 - (-0.02711/-1.68533) \approx 0.75522 - 0.01608 \approx 0.73914$
- Error = $|x_2 - x_1| = |0.73914 - 0.75522| \approx \mathbf{0.01608}$

Root Finding

Metode Newton Raphson (Contoh 1)

Iterasi 3:

- $f(0.73914) \approx -0.0001006$
- $f'(0.73914) \approx -1.67366$
- $x_3 = 0.73914 - (-0.0001006/-1.67366) \approx 0.73914 - 0.0000564 \approx 0.73908$
- Error = $|x_3 - x_2| = |0.73908 - 0.73914| \approx \mathbf{0.0000564}$

Root Finding

Metode Newton Raphson (Contoh 1)

```
# 1. Definisikan f(x) dan turunan f'(x)
f <- function(x) {
  return(cos(x) - x)
}

f_prime <- function(x) {
  return(-sin(x) - 1)
}
# 2. Inisialisasi
x_old <- 0.5     # Tebakan awal (x0)
tolerance <- 1e-7
max_iter <- 20
iter <- 0
error <- 1
# Siapkan tabel untuk hasil
results <- data.frame(Iterasi = integer(),
                      xn = double(),
                      fxn = double(), # Nilai f(x)
                      f_prime_xn = double(), # Nilai f'(x)
                      Error = double())

# 3. Proses Iterasi
while (error > tolerance && iter < max_iter) {
  iter <- iter + 1
  fx <- f(x_old)
  fpx <- f_prime(x_old)

  # Rumus Newton-Raphson
  x_new <- x_old - (fx / fpx)

  error <- abs(x_new - x_old)

  # Simpan hasil
  results[iter, ] <- c(iter, x_new, f(x_new), fpx, error)

  x_old <- x_new
}

# Tampilkan hasil
print(results, digits = 8)
cat("\nAkarnya adalah:", x_new, "\n")
```

Root Finding

Metode Newton Raphson (Contoh 1)

```
> print(results, digits = 8)
  Iterasi         xn           fxn f_prime_xn         Error
1       1 0.75522242 -2.7103312e-02 -1.4794255 2.5522242e-01
2       2 0.73914167 -9.4615381e-05 -1.6854506 1.6080751e-02
3       3 0.73908513 -1.1809779e-09 -1.6736538 5.6532229e-05
4       4 0.73908513  0.0000000e+00 -1.6736120 7.0564610e-10
> cat("\nAkarnya adalah:", x_new, "\n")

Akarnya adalah: 0.7390851
```

Root Finding

Metode Newton Raphson (Contoh 2)

$$f(x) = x^2 - x - 2.$$

Turunan: $f'(x) = 2x - 1$

Iterasi: $x_{n+1} = x_n - \frac{x_n^2 - x_n - 2}{2x_n - 1}$

$x_0 = 1.5.$

Iterasi 1:

- $f(1.5) = (1.5)^2 - 1.5 - 2 = 2.25 - 1.5 - 2 = -1.25$
- $f'(1.5) = 2(1.5) - 1 = 3 - 1 = 2$
- $x_1 = 1.5 - \frac{-1.25}{2} = 1.5 - (-0.625) = 2.125$
- Error = $|x_1 - x_0| = |2.125 - 1.5| = \mathbf{0.625}$

Root Finding

Metode Newton Raphson (Contoh 2)

Iterasi 2:

- $f(2.125) = (2.125)^2 - 2.125 - 2 = 4.515625 - 2.125 - 2 = 0.390625$
- $f'(2.125) = 2(2.125) - 1 = 4.25 - 1 = 3.25$
- $x_2 = 2.125 - \frac{0.390625}{3.25} = 2.125 - 0.12019... \approx 2.004807$
- Error = $|x_2 - x_1| = |2.004807 - 2.125| \approx \mathbf{0.120192}$

Iterasi 3:

- $f(2.004807) \approx 0.01927...$
- $f'(2.004807) \approx 3.0096...$
- $x_3 = 2.004807 - \frac{0.01927...}{3.0096...} \approx 2.004807 - 0.0064... \approx 2.00003...$
- Error = $|x_3 - x_2| \approx \mathbf{0.00477...}$

...Iterasi konvergen sangat cepat ke $x = 2$.

Root Finding

Metode Newton Raphson (Contoh 2)

Sekarang kita gunakan tebakan awal $x_0 = -0.5$ (dekat dengan -1).

Iterasi 1:

- $f(-0.5) = (-0.5)^2 - (-0.5) - 2 = 0.25 + 0.5 - 2 = -1.25$
- $f'(-0.5) = 2(-0.5) - 1 = -1 - 1 = -2$
- $x_1 = -0.5 - \frac{-1.25}{-2} = -0.5 - (0.625) = -1.125$
- Error = $|x_1 - x_0| = |-1.125 - (-0.5)| = \mathbf{0.625}$

Iterasi 2:

- $f(-1.125) = (-1.125)^2 - (-1.125) - 2 = 1.265625 + 1.125 - 2 = 0.390625$
- $f'(-1.125) = 2(-1.125) - 1 = -2.25 - 1 = -3.25$
- $x_2 = -1.125 - \frac{0.390625}{-3.25} = -1.125 - (-0.12019...) \approx -1.004807$
- Error = $|x_2 - x_1| = |-1.004807 - (-1.125)| \approx \mathbf{0.120192}$

Root Finding

Metode Newton Raphson (Contoh 2)

Iterasi 3:

- $f(-1.004807) \approx 0.01927...$
- $f'(-1.004807) \approx -3.0096...$
- $x_3 = -1.004807 - \frac{0.01927...}{-3.0096...} \approx -1.004807 + 0.0064... \approx -0.9984...$
- **Error** = $|x_3 - x_2| \approx \mathbf{0.00639...}$

...Iterasi juga konvergen sangat cepat ke $x = -1$.

Root Finding

Metode Newton Raphson (Contoh 2)

```
# 1. Definisikan f(x) dan turunan f'(x)
f <- function(x) {
  return(x^2 - x - 2)
}

f_prime <- function(x) {
  return(2*x - 1)
}

# 2. Buat fungsi solver-nya
# (Kita gabungkan tabel dan prosesnya dalam satu fungsi)
run_newton_raphson <- function(tebakan_awal) {
  x_old <- tebakan_awal
  tolerance <- 1e-7
  max_iter <- 20
  iter <- 0
  error <- 1

# Siapkan tabel untuk hasil
results <- data.frame(Iterasi = integer(),
                      xn = double(),
                      fxn = double(),
                      f_prime_xn = double(),
                      Error = double())

# Simpan kondisi awal
results[1, ] <- c(0, x_old, f(x_old), f_prime(x_old), NA)

# 3. Proses Iterasi
while (error > tolerance && iter < max_iter) {
  iter <- iter + 1
  fx <- f(x_old)
  fpx <- f_prime(x_old)

  # Hati-hati jika fpx = 0 (turunan nol)
  if (abs(fpx) < 1e-10) {
    cat("Turunan nol, metode gagal.\n")
    break
  }
```

Root Finding

Metode Newton Raphson (Contoh 2)

```
x_new <- x_old - (fx / fpx)
error <- abs(x_new - x_old)

  results[iter + 1, ] <- c(iter, x_new, f(x_new), f_prime(x_new), error)
x_old <- x_new
}

# Tampilkan hasil
print(results, digits = 8)
cat("\nAkarnya adalah:", x_new, "\n")
cat("------------------------------------------------\n")
}
```

```
# --- JALANKAN SKENARIO ---

cat("Mencari akar dari f(x) = x^2 - x - 2\n")
cat("------------------------------------------------\n")

cat("\nSKENARIO A: Tebakan awal x0 = 1.5 (Target 2)\n")
run_newton_raphson(1.5)

cat("\nSKENARIO B: Tebakan awal x0 = -0.5 (Target -1)\n")
run_newton_raphson(-0.5)
```

Root Finding

Metode Newton Raphson (Contoh 2)

```
SKENARIO A: Tebakan awal x0 = 1.5 (Target 2)
> run_newton_raphson(1.5)
  Iterasi        xn           fxn f_prime_xn        Error
1       0 1.5000000 -1.2500000e+00  2.0000000           NA
2       1 2.1250000  3.9062500e-01  3.2500000 6.2500000e-01
3       2 2.0048077  1.4446191e-02  3.0096154 1.2019231e-01
4       3 2.0000077  2.3040118e-05  3.0000154 4.8000123e-03
5       4 2.0000000  5.8982152e-11  3.0000000 7.6800000e-06
6       5 2.0000000  0.0000000e+00  3.0000000 1.9660717e-11

Akarnya adalah: 2
```

```
SKENARIO B: Tebakan awal x0 = -0.5 (Target -1)
> run_newton_raphson(-0.5)
  Iterasi         xn           fxn f_prime_xn        Error
1       0 -0.5000000 -1.2500000e+00 -2.0000000           NA
2       1 -1.1250000  3.9062500e-01 -3.2500000 6.2500000e-01
3       2 -1.0048077  1.4446191e-02 -3.0096154 1.2019231e-01
4       3 -1.0000077  2.3040118e-05 -3.0000154 4.8000123e-03
5       4 -1.0000000  5.8982152e-11 -3.0000000 7.6800000e-06
6       5 -1.0000000  0.0000000e+00 -3.0000000 1.9660717e-11

Akarnya adalah: -1
```

SEE YOU NEXT WEEK !

Ferdian Bangkit Wijaya, S.Stat., M.Si
NIP. 199005202024061001
ferdian.bangkit@untirta.ac.id