

LAPORAN ANALISA HASIL PROGRAM REGULA FALSI

SEMESTER GANJIL (3)

2020 / 2021



NAMA : AKBAR UMAR ALFAROQ

NPM : 19081010043

FAKULTAS : ILMU KOMPUTER

PROGRAM STUDI : TEKNIK INFORMATIKA

UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"

JAWA TIMUR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Pada kesempatan kali ini saya akan menjelaskan step by step hasil analisa program *Regula Falsi* yang telah saya buat menggunakan bahasa pemrograman **Python** menggunakan Jupyter Notebook. Jupyter Notebook adalah platform Notebook yang didalamnya bisa menjalankan kode program **Python**, format markdown, visualisasi data dll. Sehingga memudahkan user dalam membacanya (seperti membaca *e-book*).

Langsung saja ke laporan analisa

Ketika program pertama dijalankan, program akan meminta user untuk menginputkan batas atas dan batas bawah / *range*

Masukkan batas bawah : 1

Masukkan batas atas :

2

Berikut Algoritmanya

```
# Input Data
while True:
    a = float(input("Masukkan batas bawah : "))
    b = float(input("Masukkan batas atas : "))

    # Melakukan pengecekan, apakah diantara range tersebut terdapat akar
    if (f(a) * f(b) > 0):
        print("Diantara range yang diinputkan tidak terdapat akar, harap coba lagi\n")
    else:
        break
```

Algoritma diatas adalah seperti melakukan *do-while* pada C/C++, Disana user melakukan input angka, lalu dilakukan validasi apakah diantara range tersebut terdapat akar, jika ya maka selesai, jika tidak maka muncul pesan validasi dan user akan diminta untuk menginputkan ulang range-nya

Masukkan batas bawah : 1

Masukkan batas atas : 1.5

Diantara range yang diinputkan tidak terdapat akar, harap coba lagi

Masukkan batas bawah :

Jika user menginputkan range dengan benar, maka selanjutnya akan diminta menginputkan toleransi error dan maximum jumlah iterasi

Masukkan batas bawah : 1

Masukkan batas atas : 2

Masukkan toleransi error : 0.000000001

Masukkan batas max. iterasi :

15

Berikut algoritmanya

```
# Input toleransi eror dan max. iterasi
errorTolerant = float(input("Masukkan toleransi error : "))
maxIteration = int(input("Masukkan batas max. iterasi : "))
print("\n")
```

Setelah user menginputkan data dengan benar, maka program akan menjalankan algoritma regula falsi dibawah ini

```
# Agar dinamis, kita menggunakan paramter agar fungsi re-usable
def regulaFalsi(a, b, errorTolerant, maxIteration):
    c = 0
    printKolom()
    # Iterasi sebanyak jumlah max. iterasi
    for i in range(maxIteration):
        c = (f(b)*a - f(a)*b) / (f(b) - f(a));
        Fc = f(c);
        # Tampilkan Value
        print("%i.\t%.9f\t%.9f\t%.9f\t%.9f\t%.9f\n" % (i+1, a, c, b, f(a), Fc));

        if (f(a) * Fc > 0):
            a = c;
        else:
            b = c;

        if (Fc <= errorTolerant): break

    # Kembalikan nilai fungsi
    return c
```

Pada fungsi diatas menerima 4 parameter yaitu a = batas bawah, b = batas atas, $errorTolerant$, dan $maxIteration$, yang mana masing-masing berfungsi sesuai dengan namanya. Pada permulaan fungsi tersebut memanggil "printKolom()" yang mana untuk menampilkan kolom2 atas, berikut pendefinisianya beserta $f(x)$.

```
def f(x):
    return math.pow(x, 2) - math.exp(x) + 3

def printKolom():
    print("No.\ta\tc\tb\tF(a)\tF(c)")
```

Ini adalah hasil dari "printKolom"

```
Masukkan batas bawah : 1
Masukkan batas atas : 2
Masukkan toleransi error : 0.000000001
Masukkan batas max. iterasi : 15
```

No.	a	c	b	F(a)	F(c)
-----	---	---	---	------	------

Setelah itu pada bagian selanjutnya di bawah ini, melakukan iterasi sebanyak max. Iterasi yang diinputkan user. Lalu menghitung nilai c , $F(c)$, lalu menampilkannya. Setelah itu dilakukan pengecekan mana yang memiliki tanda yang sama dengan $F(c)$ lalu nilainya akan digeser ke c . Lalu dilakukan pengecekan apakah $F(c)$ sudah memenuhi eror yang ditoleransi, jika iya maka berhenti, jika tidak maka perulangan berlanjut.

```
# Agar dinamis, kita menggunakan paramter agar fungsi re-usable
def regulaFalsi(a, b, errorTolerant, maxIteration):
    c = 0
    printKolom()
    # Iterasi sebanyak jumlah max. iterasi
    for i in range(maxIteration):
        c = (f(b)*a - f(a)*b) / (f(b) - f(a));
        Fc = f(c);
        # Tampilkan Value
        print("%i.\t%.9f\t%.9f\t%.9f\t%.9f\t%.9f\n" % (i+1, a, c, b, f(a), Fc));

        if (f(a) * Fc > 0):
            a = c;
        else:
            b = c;

        if (Fc <= errorTolerant): break

    # Kembalikan nilai fungsi
    return c
```


Dan yang terakhir adalah mengembalikan nilai akar itu sendiri

```
# Agar dinamis, kita menggunakan paramter agar fungsi re-usable
def regulaFalsi(a, b, errorTolerant, maxIteration):
    c = 0
    printKolom()
    # Iterasi sebanyak jumlah max. iterasi
    for i in range(maxIteration):
        c = (f(b)*a - f(a)*b) / (f(b) - f(a));
        Fc = f(c);
        # Tampilkan Value
        print("%i.\t%.9f\t%.9f\t%.9f\t%.9f\t%.9f\n" % (i+1, a, c, b, f(a), Fc));

        if (f(a) * Fc > 0):
            a = c;
        else:
            b = c;

        if (Fc <= errorTolerant): break

    # Kembalikan nilai fungsi
    return c
```

Output dari program diatas adalah seperti ini

```
Masukkan batas bawah : 1
Masukkan batas atas : 2
Masukkan toleransi error : 0.000000001
Masukkan batas max. iterasi : 15
```

No.	a	c	b	F(a)	F(c)
1.	1.000000000	1.767140238	2.000000000	1.281718172	0.268696519
2.	1.767140238	1.862265070	2.000000000	0.268696519	0.029727714
3.	1.862265070	1.872042298	2.000000000	0.029727714	0.002981391
4.	1.872042298	1.873015399	2.000000000	0.002981391	0.000295957
5.	1.873015399	1.873111923	2.000000000	0.000295957	0.000029349
6.	1.873111923	1.873121494	2.000000000	0.000029349	0.000002910
7.	1.873121494	1.873122443	2.000000000	0.000002910	0.000000289
8.	1.873122443	1.873122537	2.000000000	0.000000289	0.000000029
9.	1.873122537	1.873122547	2.000000000	0.000000029	0.000000003
10.	1.873122547	1.873122548	2.000000000	0.000000003	0.000000000

Akar dari fungsi $x^2 - \exp^x + 3$ adalah : 1.873122548

Untuk menjalankan program notebook ini bisa menggunakan Jupyter Notebook (lokal), atau jika online bisa menggunakan Google Colab <https://colab.research.google.com/notebooks/>, lalu upload file .ipynb nya

Cukup sekian yang bisa saya sampaikan untuk Analisa Hasil Program Regula Falsi ini, jika ada kekurangan saya mohon maaf.

Terima Kasih Bu Yisti atas materi yang telah disampaikan tentang Regula Falsi ini, Sekian dan Terima Kasih

Wassalamu'alaikum Warahmatullahi Wabarakatuh