

LAPORAN

IMPLEMENTASI METODE TABEL DALAM PEMROGRAMAN

Hamdan Yuwafi Mastu Wijaya
(NPM. 19081010092)

A. Pendahuluan

Metode numerik merupakan teknik penyelesaian permasalahan yang diformulasikan secara matematis dengan menggunakan operasi hitungan (aritmatik) yaitu operasi tambah, kurang, kali, dan bagi. Metode ini digunakan karena banyak permasalahan matematis tidak dapat diselesaikan menggunakan metode analitik. Jikapun terdapat penyelesaiannya secara analitik, proses penyelesaiannya sering kali cukup rumit dan memakan banyak waktu sehingga tidak efisien.

Dalam menentukan penyelesaian persamaan non-linier dengan metode numerik, terdapat dua metode yang dapat digunakan, yaitu metode terbuka dan metode tertutup (*bracketing*).

Dalam metode terbuka, terdapat tiga sub-metode, yaitu metode iterasi sederhana, metode newton-raphson dan metode secant. Sedangkan dalam metode tertutup juga terdapat tiga sub-metode, yaitu metode tabel, metode biseksi dan metode regula falsi. Laporan ini berfokus pada metode tertutup dengan sub-metode tabel.

Prinsip dari metode ini adalah x di antara a dan b dibagi sebanyak N bagian dan pada masing-masing bagian dihitung nilai $f(x)$ sehingga diperoleh tabel :

x	$f(x)$
$x_0 = a$	$f(x_0)$
x_1	$f(x_1)$
x_2	$f(x_2)$
...	...
x_n	$f(x_n)$

B. Algoritma

1. Mendefinisikan nilai $f(x)$;
2. Menentukan range x (batas atas (a) dan batas bawah (b) dari x);
3. Menentukan N sebagai pembagi (ketelitian);
4. Menghitung step pembagi h :
$$h = \frac{a-b}{N}$$
5. Melakukan operasi perulangan dari $i = 0$ hingga N :
$$x_i = b + i.h$$
$$y_i = f(x_i)$$
6. Untuk $i = 0$ hingga N , dimana:
 - a. Bila $f(x) = 0$, maka akarnya x_k
 - b. Bila $f(a) \cdot f(b) < 0$, maka:
 - i. $f(x_k) \leq f(x_{k+1})$, maka akarnya x_k
 - ii. Bila tidak, x_{k+1} adalah penyelesaian atau dapat dikatakan penyelesaian berada di antara x_k dan x_{k+1} .

C. Implementasi

a. Struktur File Aplikasi

```
+ table
  - core.js
  - index.html
  - logic.js
```

File *core.js* memuat kelas utama dan rumus persamaan untuk melakukan perhitungan matematika. File *logic.js* memuat kontrol input dari form html dan memproses hasil perhitungan dari *core.js* dan mengembalikan hasilnya untuk ditampilkan di file html. File *index.html* merupakan interface aplikasi yang berisi form bahan perhitungan dan tempat keluarnya hasil.

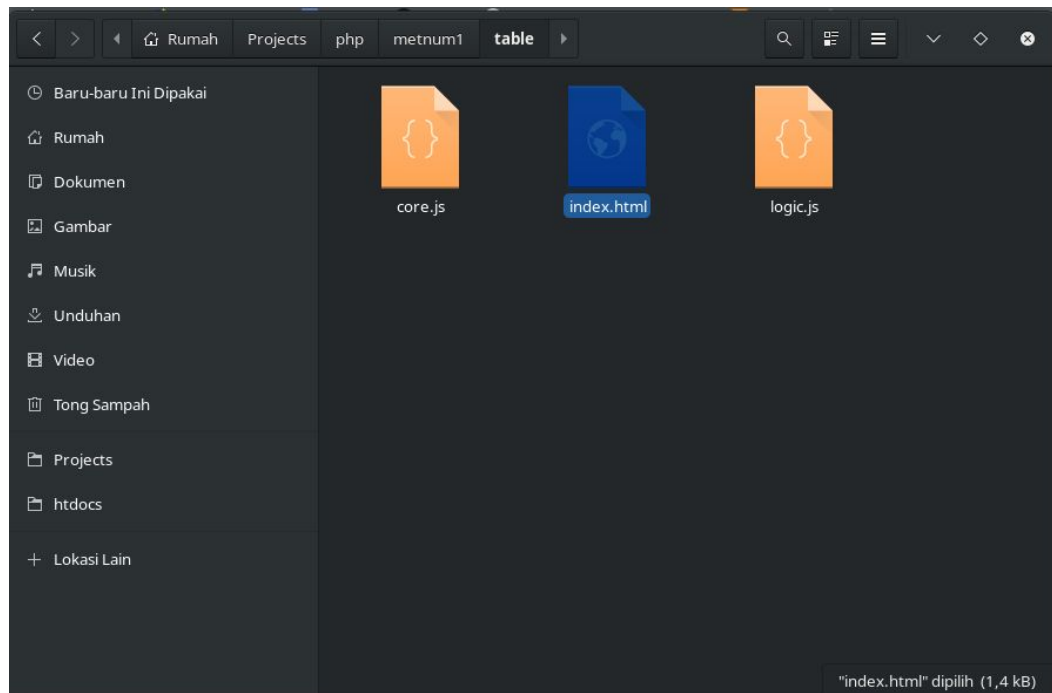
b. Source Code utama

```
process() {
    var count = Math.abs((this.va - this.vb) / this.vt)
    let ress = []; let data = []
    for (let i = 0; i <= count; i++) {
        data[i] = this.va - (this.vt * i)
        ress[i] = f(data[i])
    }

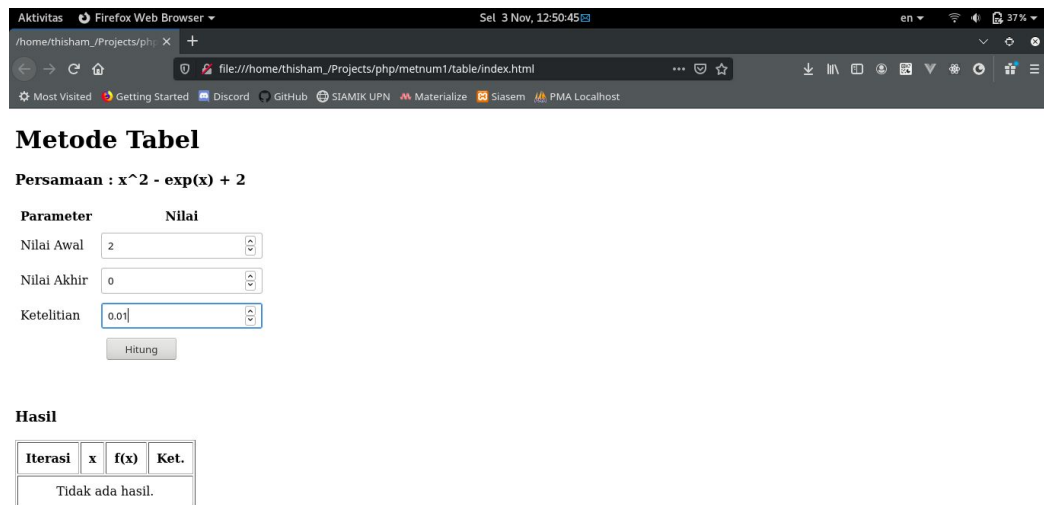
    this.data = data
    this.ress = ress
    return this
}
```

D. Dokumentasi

1. Buka folder program, pilih index.html dan buka di browser;



3. Setelah terisi, Tekan tombol hitung;



Metode Tabel

Persamaan : $x^2 - \exp(x) + 2$

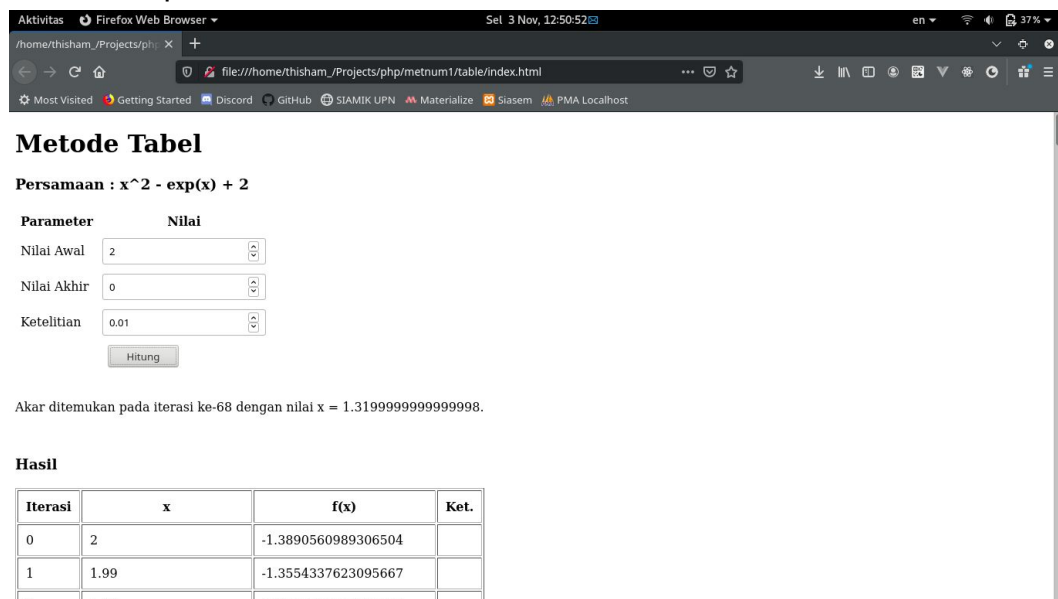
Parameter	Nilai
Nilai Awal	2
Nilai Akhir	0
Ketelitian	0.01

Hitung

Hasil

Iterasi	x	f(x)	Ket.
Tidak ada hasil.			

4. Periksa hasilnya, bila akar ditemukan, akan muncul nilai akar dan pada iterasi berapa ia ditemukan.



Metode Tabel

Persamaan : $x^2 - \exp(x) + 2$

Parameter	Nilai
Nilai Awal	2
Nilai Akhir	0
Ketelitian	0.01

Hitung

Akar ditemukan pada iterasi ke-68 dengan nilai $x = 1.3199999999999998$.

Hasil

Iterasi	x	f(x)	Ket.
0	2	-1.3890560989306504	
1	1.99	-1.3554337623095667	

E. Analisa

Dari percobaan di atas, akar didapatkan dari nilai x yang bernilai $f(x)$ paling mendekati nol. Namun, program ini memiliki kekurangan, yaitu tidak dapat menentukan nilai x yang berada di antara x_i dan x_{i+1} .

F. Kesimpulan

Setiap variabel yang diinputkan dalam percobaan yang telah dilakukan mempengaruhi nilai dari toleransi error, baik itu jumlah iterasi, batas atas, maupun batas bawah. Semakin banyak jumlah iterasi maka toleransi error akan semakin mendekati 0. Demikian pula untuk range antara batas bawah dan batas atas, yang apabila semakin kecil maka toleransi toleransi error akan semakin mendekati 0.