

THỰC HÀNH PHƯƠNG PHÁP TÍNH

PHƯƠNG TRÌNH VÀ HỆ PHƯƠNG TRÌNH PHI TUYẾN

Đề bài

Giải phương trình

$$f(x) = \tan(x) - x - 0,5 = 0 \quad (1)$$

Lần lượt giải bài toán bằng các phương pháp sau:

1. *Phương pháp chia đôi khoảng cách*, sao cho sai số $\text{eps} = 0.00001$.
2. *Phương pháp nội suy tuyến tính*, sao cho $f(x_r) \leq \text{eps} = 0.00001$.
3. *Phương pháp lặp*, sao cho $|x_{i+1} - x_i| \leq \text{eps} = 0.00001$.
4. *Phương pháp Newton*, sao cho $|x_{i+1} - x_i| \leq \text{eps} = 0.00001$.

HƯỚNG DẪN

Đầu tiên, ta thực hiện các bước sau:

B1. Mở file mới, viết hàm $f(x) = \tan(x) - x - 0,5$ (hàm này sẽ dùng chung cho câu 1, 2)

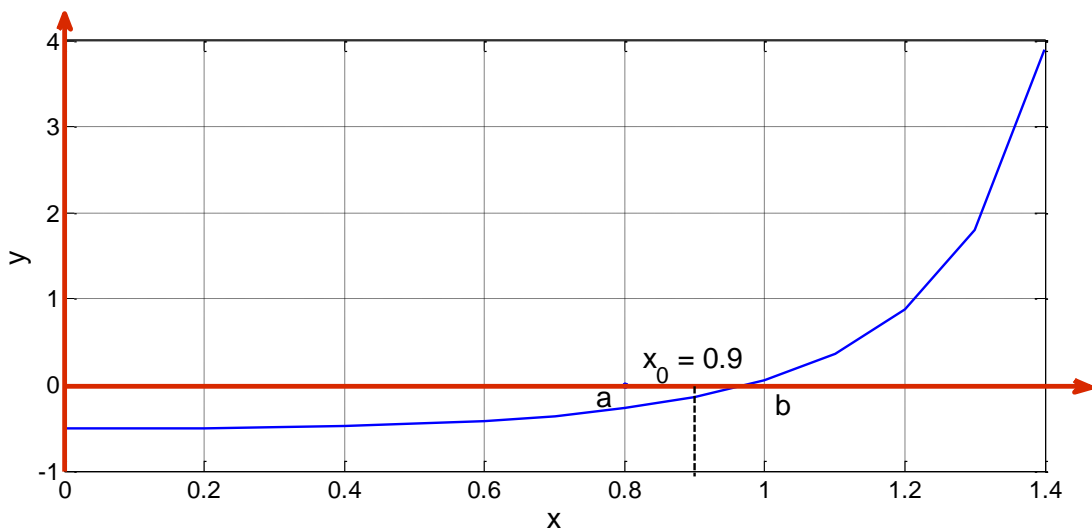
```
function y = f1(x)
    y = tan(x) - x - 0.5;
end
```

B2. Vẽ đồ thị $y = f(x)$ để:

- xác định khoảng $[a, b]$ chứa nghiệm (dùng cho câu 1, 2)
- xác định nghiệm thô x_0 (dùng cho câu 3, 4)

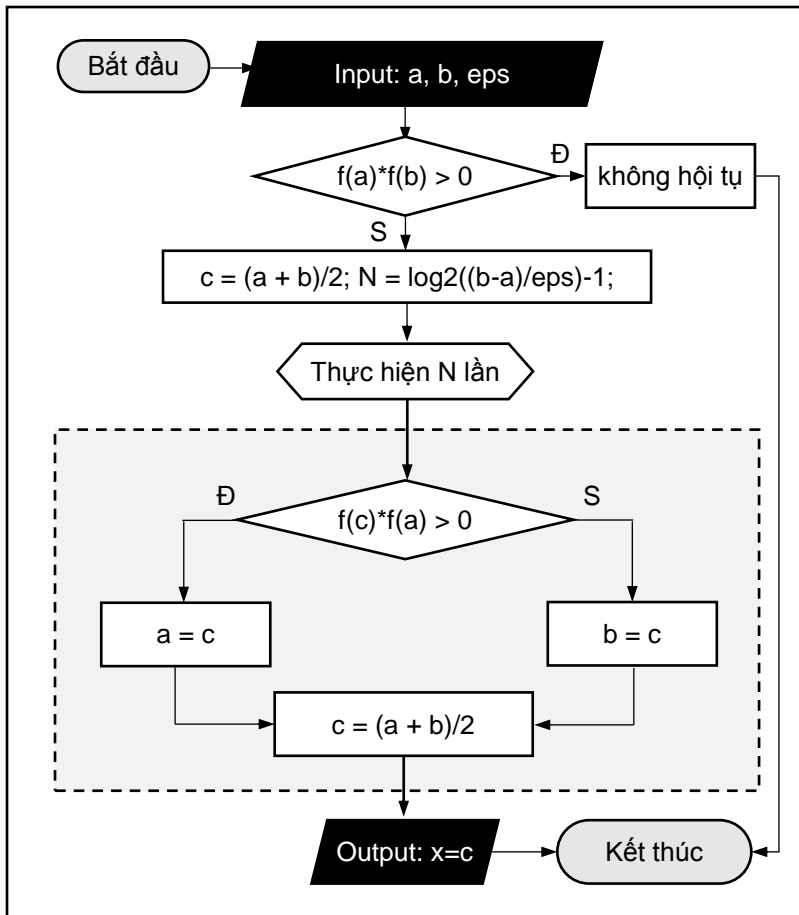
Tại giao diện command window, gõ:

```
>> x = 0:0.1:1.4;      %Tạo mảng x
>> y = f1(x);          %Gọi hàm f1 vừa viết để xác định các giá trị của y theo x
>> plot(x, y); grid('on'); %Dùng hàm plot để vẽ đồ thị; hàm grid để hiển thị các đường kẻ ô
```



Nhận xét, dựa vào đồ thị, ta có thể lấy $a = 0.8$; $b = 1$; và $x_0 = 0.9$

B3. Lần lượt giải phương trình trên bằng 4 phương pháp mà đề yêu cầu.



1. Phương pháp chia đôi khoảng cách

1.1. Phân tích bài toán

- **Xác định Input và Output**

- Input: a, b, eps.
- Output: x

- **Tính số lần lặp**

Dựa vào công thức tính sai số

$$\text{esp} < \frac{b-a}{2^{N+1}}.$$

suy ra số lần lặp :

$$N < \log_2((b-a)/\text{eps}) - 1.$$

- **Thuật toán**

Xem lưu đồ hình bên

1.2. Viết hàm

```

function x = chia_doi_khoang(a,b,eps)
if f1(a)*f1(b) > 0 %kiểm tra điều kiện hội tụ của bài toán
    x = 'bài toán không hội tụ';
else
    c = (a+b)/2;
    N = log2((b-a)/eps)-1
    for i=1:N
        if f1(c)*f1(a) > 0
            a = c;
        else
            b = c;
        end
        c = (a + b)/2;
    end
    x = c;
end
end
  
```

2. Phương pháp nội suy tuyến tính

2.1. Phân tích bài toán

- **Xác định Input và Output**

- Input: a, b, eps.
- Output: x.

- **Thuật toán**

- Cho một hàm $f(x)$ liên tục trong đoạn $[a, b]$ sao cho $f(a)f(b) < 0$.
- Thực hiện:

$$(1) \quad c = \frac{a.f(b) - b.f(a)}{f(b) - f(a)}$$

(2) Nếu $f(a).f(c) > 0$ thì $a = c$; ngược lại thì $b = c$.

cho đến khi $f(x) \leq \text{esp}$ thì dừng. Khi đó nghiệm của phương trình $x = c$.

2.2. Viết hàm

```
function x = noi_suy_tt(a,b,eps)
if f1(a)*f1(b)>0
    x = 'bai toan khong hoi tu';
else
    c = (a*f1(b) - b*f1(a))/(f1(b)-f1(a));
    while abs(f1(c))>eps
        c = (a*f1(b) - b*f1(a))/(f1(b)-f1(a));
        if f1(a)*f1(c)>0
            a=c;
        else
            b=c;
        end
    end
    x=c;
end
end
```

3. Phương pháp lặp

3.1. Phân tích bài toán

- **Xác định Input và Output**

- Input: x_0 , ϵ .
- Output: x

- **Thuật toán**

- Xác định $x = g(x)$ sao cho $|g'(x)| < 1$
- Gán $x_1 = g(x_0)$.
- Thực hiện:

$$x_0 = x_1$$

$$x_1 = g(x_0)$$

cho đến khi $|x_1 - x_0| \leq \epsilon$ thì dừng. Khi đó nghiệm của phương trình $x = x_1$.

3.2. Viết hàm

- Ngoài hàm chính, ta cần 2 hàm con là hàm tính $g(x)$ và hàm tính đạo hàm $g'(x)$. Do đó, ta mở 2 file mới và viết các hàm sau:

- Hàm $g(x)$

```
function y = g1(x)
y = atan(x+0.5);
end
```

- Đạo hàm $g'(x)$

```
function y = daoham_g1(x)
y=1/(x^2+1);
end
```

- Hàm chính:

```
function x=lap(x0,eps)
if abs(daoham_g1(x0))<1    %Kiểm tra điều kiện hội tụ của bài toán
    x1=g1(x0);
    while abs(x1-x0)>eps
        x0=x1;
        x1=g1(x0);
    end
    x=x1;
else
    x = 'bai toan khong hoi tu';
end
end
```

4. Phương pháp Newton

4.1. Phân tích bài toán

- **Xác định Input và Output**

- Input: x_0 , ϵ .
- Output: x .

- **Thuật toán**

- Kiểm tra điều kiện hội tụ: $\left| \frac{f(x_0) \cdot f''(x_0)}{(f'(x_0))^2} \right| < 1$
- Nếu điều kiện hội tụ được thỏa, thực hiện

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

$$x_0 = x_1$$

cho đến khi $|x_1 - x_0| \leq \epsilon$. Khi đó, nghiệm của phương trình $x = x_1$.

4.2. Viết hàm

- Ngoài hàm chính, ta cần 3 hàm con:

- hàm $f(x)$ (đã viết ở trên).
- hàm $f'(x)$.
- hàm $f''(x)$.

Hàm tính đạo hàm bậc nhất

```
function y = daoham1(x)
y = 1/(cos(x))^2 - 1;
end
```

Hàm tính đạo hàm bậc hai

```
function y = daoham2(x)
y = 2*sin(x)/(cos(x))^3;
end
```

- **Hàm chính**

```
function x = Newton(x0,eps)
if abs( f1(x0)*daoham1(x0) / (daoham2(x0))^2 ) > 1
    x = 'Bai toan khong hoi tu;
else
    x1=x0-f1(x0)/daoham1(x0);
    while abs(x1-x0)>eps
        x0=x1;
        x1=x0-f1(x0)/daoham1(x0);
    end
    x=x1;
end
end
```

GỌI HÀM VÀ KẾT QUẢ

Gán giá trị cho các tham số

```
>> eps = 0.0001; a = 0.8; b = 1; x0 = 0.9;
```

1. Phương pháp chia đôi khoảng cách:

```
>> x = chia_doi_khoang(a,b,eps)
```

Kết quả: $x = 0.975017193264126$

2. Phương pháp nội suy tuyến tính:

```
>> x = noi_suy_tt(a,b,eps)
```

Kết quả: $x = 0.975017193264127$

3. Phương pháp lặp:

```
>> x = lap(x0,eps)
```

Kết quả: $x = 0.975017193264126$

4. Phương pháp Newton:

BÀI TẬP VỀ NHÀ

Cho phương trình

$$x^3 - 2x - 10 = 0 \quad (2)$$

1. Vẽ đồ thị tìm đoạn $[a, b]$ chứa nghiệm và nghiệm nhỏ x_0 .
2. Viết chương trình (hàm) thực hiện các nhiệm vụ sau:
 - a. Yêu cầu người dùng nhập một số nguyên tùy ý. Lưu số này vào biến ch.
 - b. Tùy vào giá trị của ch, thực hiện các nhiệm vụ sau:
 - Nếu ch = 1: Hiện thông báo '*Phương pháp chia đôi khoảng cách*' và giải phương trình bằng phương pháp chia đôi khoảng cách. *Sau đó, yêu cầu người dùng nhập lại i ch.*
 - Nếu ch = 2: Hiện thông báo '*Phương pháp lặp p*' và giải phương trình bằng phương pháp lặp p. *Sau đó, yêu cầu người dùng nhập lại i ch.*
 - Nếu ch = 3: Hiện thông báo '*Phương pháp Newton*' và giải phương trình bằng phương pháp Newton. *Sau đó, yêu cầu người dùng nhập lại i ch.*
 - Nếu ch = 0: Thoát.
 - Nếu ch $\notin \{0, 1, 2, 3\}$: Thông báo yêu cầu người dùng nhập lại i ch.