MATLAB 언어특징 20150339 김진민

A(□,△) ⇒ □행 △열 원소를 의미

A(□) ⇒ 순서대로 □번째에 원소를 의미

a:b:c ⇒ a부터 c까지 b간격으로 (생략시 b=1)

[출력 변수들] = function_name (입력변수들)

...

end

⇒ z = fun(3,7)로 호출

※ New-Rap 법 코드 예시

```
iter = 0;
while (1)
  xrold = xr;
  xr = xr - func(xr)/dfunc(xr);   % 손미분
  iter = iter+1;
  if xr ~= 0
    ea = abs((xr-xrold)/xr)*100;
  end
  if ea <= es || iter >= maxit
    break;
  end
end-while
root = xr;
end-func
```

9장) Gauss 소거법 → 손으로만

- 피봇원소가 0인 경우 피봇팅 (0에 매우가까운경우도)

(1) $①_2' = ① \times \frac{a_{21}}{a_{11}}$

(2) $②' = ② - ①_2'$

(3) $①_3' = ① \times \frac{a_{31}}{a_{11}}$

(4) $③' = ③ - ①_3'$

(5) $②_3'' = ②' \times \frac{a_{32}'}{a_{22}'}$

(6) $③'' = ③' - ②_3''$

(7) ③″식 → x_3 도출

②′식 → x_2 도출

① 식 → x_1 도출

수치오차

① 반올림 오차

⇒ 컴퓨터상의 수 표현으로 발생

⇒ 뺄셈의 무효화

② 절단오차

⇒ 수학적연산을 근사표현해서 발생

ex) Taylor 급수

0차근사: $f(x_{i+1}) \cong f(x_i)$

1차근사: $f(x_{i+1}) \cong f(x_i) + f'(x_i)h$

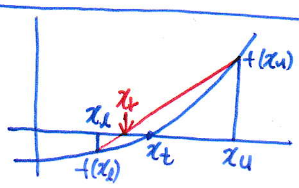
2차근사: $f(x_{i+1}) \cong f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2}h^2$

※ 할선법 공식

$$x_{i+1} = x_i - \frac{f(x_i)\boxed{(x_{i-1}-x_i)}}{\boxed{f(x_{i-1})-f(x_i)}}$$

$$x_{i+1} = x_i - \frac{f(x_i)\boxed{\delta x_i}}{\boxed{f(x_i+\delta x_i)-f(x_i)}}$$

Point! $f'(x_i) \cong \frac{f(x_{i-1})-f(x_i)}{x_{i-1}-x_i}$

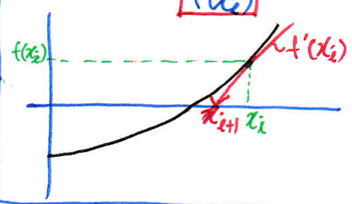


※ 가위치법 공식

$$x_r = x_u - \frac{f(x_u)(x_l - x_u)}{f(x_l)-f(x_u)}$$

※ New-Rap 법 공식

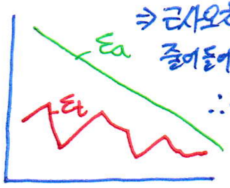
$$x_{i+1} = x_i - \frac{f(x_i)}{\boxed{f'(x_i)}}$$



※ 이분법 코드 예시

```
iter = 0;
xr = xl;
while (1)
  xrold = xr;
  xr = (xl+xu)/2;
  iter = iter+1;
  disp(iter); disp(xr);
  if xr ~= 0
    ea = abs((xr-xrold)/xr)*100;
  end
  test = feval(func,xl)
       *feval(func,xr);
  if test < 0
    xu = xr;
  elseif test > 0
    xl = xr;
  else
    ea = 0
  end
  if ea <= es || iter >= maxit
    break, end
end
```

※ 이분법의 오차



⇒ 근사오차가 늘 쭉 줄어들어 예측할수있음!

∴ 근사오차를 씀.

"n번 반복후 오차가 $E_{a,d}$라 할때" $n = \log_2(\Delta x^0 / E_{a,d}) - 1$

4장)	절대 오차	상대오차(%)
<참값을 알때> 참오차 (True...)	E_t = 참값-근사값	$\varepsilon_t = \frac{\text{참값-근사값}}{\text{참값}} \times 100(\%)$
<참값을 모를때> 근사오차 (Approxi...)	E_a = 현재근사값 -이전근사값	$\varepsilon_a = \frac{\text{현재근사값-이전근사값}}{\text{현재근사값}} \times 100(\%)$

※ ε_s (백분율 허용치) = $(0.5 \times 10^{2-n})\%$ →

5~6장) 코딩, 손계산

초기가정법
- 1) 구간법 (- 양끝값가정, - 수렴느림, - 항상근찾음)
  - ① 증분탐색법 (강조X) ~~[illegible]~~
  - ② 이분법
  - ③ 가위치법
- 2) 개방법 (- x값 가정, - 수렴빠름, - 근을못찾을수도)
  - ① 단순고정점 반복법 (강조X) $g(x) = f(x) + x$, $x_{i+1} = g(x_i)$
  - ② Newton - Raphson 법
  - ③ 할선법 (+수정된 할선법)

ex 4.1 코드

```
n=3;
clear;
Es = (0.5*10^(2-n));
x=0.5;
Exp_x=0;
disp('항수 결과 오차 근사오차');
for i=1:1:6
  Exp_x = Exp_x + x^(i-1)/factorial(i-1);
  Er_t = (exp(0.5) - Exp_x)/exp(0.5)*100;
  if i>1
    Er_s = (Exp_x - Exp_x_old)/Exp_x*100;
  end
  Exp_x_old = Exp_x;
```

↓ 계속

```
  if i==1
    fprintf('%d %f %f\n', i,
            Exp_x, Er_t)
  else
    fprintf('%d %f %f %f\n', i,
            Exp_x, Er_t, Er_s)
  end
end
```