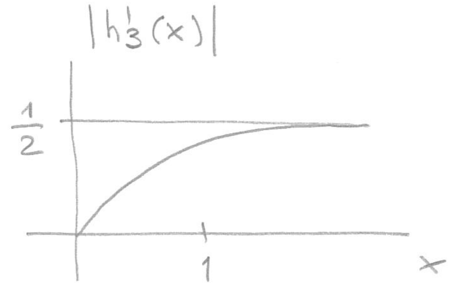
(5) h_3 utilizzabile ?

$$h_3'(x) = \frac{1-e^{-x}}{2}$$



Siccome $\alpha \in (0,1)$,

CERTAMENTE $|h_3'(x)| < 1 \Rightarrow \exists$ int...

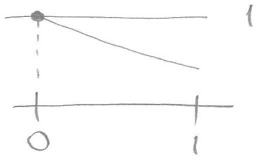
Dunque: h_3 UTILIZZABILE per approx α.

(6) Come determino x_0 (ad es per m. it def da h_2):

- cerco int che verifica ip (1) e (2) del Teo conv...

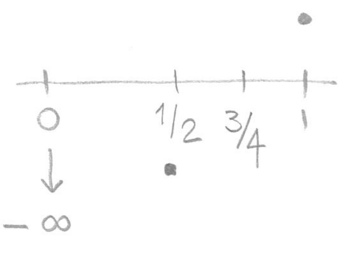
... $(0,1)$ non va bene perché non chiuso

... si ha:



, quindi va certam bene un int del tipo $[a,1]$ con $[a,1] \ni \alpha$. Lo cerco con bisezione:

$$f\left(\tfrac{1}{2}\right) = \tfrac{1}{2} - \ln 2 < 0$$



$\Rightarrow \left[\frac{1}{2}, 1\right]$ ok!

Per det x_0 ("l'estremo più vicino ad α") devo decidere il segno di $f\left(\frac{3}{4}\right)$...

(7) Quanto rapidam converge la succes ?

- studio quanto rapidam $x_k - \alpha \to 0$;
- $x_k - \alpha = h(x_{k-1}) - h(\alpha) \overset{\text{se } h \in \mathcal{C}^1[a,b]}{=} h'(\theta_k)(x_{k-1} - \alpha)$

θ_k tra x_{k-1} e α

ovvero, posto $d_k = |x_k - \alpha|$:

$$d_k = |h'(\theta_k)|\, d_{k-1}$$

- Siccome $\boxed{h'(x) \neq 0 \text{ su } [a,b]}$ e $d_0 > 0$,
  allora: $\forall k,\ d_k > 0$
- $$\lim_{k \to \infty} \frac{d_k}{d_{k-1}} = \lim_{k \to \infty} |h'(\theta_k)| = |h'(\alpha)|$$
- d_k si comporta, "asintoticamente", come la succes $|h'(\alpha)|^k d_0$.
- Nel caso del m. it def da h_2: $|h_2'(\alpha)| \simeq 0{,}57$
- Nel caso del m. it def da h_3: $|h_3'(\alpha)| \simeq 0{,}25$

```
//
// LMV_MetodiUnPunto_00.sce (reperibile sulla pagina web del corso)
//
// Metodo ad un punto definito dalla funzione h (con derivata continua)
//
function y=h(x)
    y = exp(-x)
endfunction
//
function j=dh(x)
    j = -exp(-x)
endfunction
//
// Grafici iniziali
//
clf();
subplot(211);
xx = linspace(-1,3,200);
plot(xx,h(xx),"b",xx,xx,"k");
xlabel("x");
legend("y = h(x)","y = x");
xgrid();
subplot(212);
plot(xx,abs(dh(xx)),"b");
xlabel("x");
ylabel("|h''(x)|")
xgrid();
//
x0 = input("scelta punto iniziale: x0=");
//
// Iterazione
//
x_mit = x0;
tol = 1d-6; // Errore assoluto richiesto.
kmax = 30;
StimaErr = []; // Riga delle stime di errore.
info = -1; // Flag per decidere se interrompere l'iterazione.
k = 0;
//
while info == -1,
    if k > kmax then info = 3; z = x_mit; break; end;
    if dh(x_mit) == 1 then info = 2; z = []; end;
    StimaErr($+1) = abs((h(x_mit) - x_mit)/(dh(x_mit) - 1));
        // StimaErr: vedi nota finale.
    if StimaErr($) < tol then info = 1; z = x_mit; break; end;
    x_mit = h(x_mit);
    k = k+1;
end;
//
// Fine iterazione
//
printf("\nz ="); disp(z);
printf("\nnumero iterazioni ="); disp(k);
printf("\ninfo ="); disp(info);
//
// Grafici
//
clf();
subplot(121);
plot(xx,h(xx),"b",xx,xx,"k",z,h(z),"ro");
xlabel("x");
legend("y = h(x)","y = x","(z,h(z))");
```

```
xgrid();
subplot(122);
plot2d(log10(StimaErr(1:$-1)),log10(StimaErr(2:$)),5,frameflag=4);
xlabel("log(e(k))");
ylabel("log(e(k+1))");
xgrid();
//
// Se h ha derivata prima continua allora, detto P il punto unito di h:
//
//   h(x_mit) - x_mit    h(x_mit) - h(P) - (x_mit - P)
//   ---------------- =  ----------------------------- =
//    h'(x_mit) - 1             h'(x_mit) - 1
//
//       h'(t)(x_mit - P) - (x_mit - P)    (h'(t) - 1) (x_mit - P)
//     = ------------------------------ =  -----------------------
//               h'(x_mit) - 1                  h'(x_mit) - 1
//
// con t tra x_mit e P (Teorema di Lagrange). SE la successione
// generata converge a P, allora anche t -> P e:
//
//            h'(t) - 1
//          -------------  ->  1
//          h'(x_mit) - 1
//
// Quando x_mit è vicino a P, lo stimatore fornisce valori affidabili.
//
```