Esercizi svolti in aula (lezioni del 28 aprile e del 2 maggio 2008)

Esercizio 1
La funzione

$$f(x) = \frac{\sqrt{1-\cos^2 x}}{\sin x}$$

è identicamente uguale a 1. Tuttavia, in aritmetica finita, per valori di x abbastanza piccoli, si verifica una forte cancellazione nel calcolo del radicando. Con un grafico si mette in evidenza il fenomeno per $10^{-8} \le x \le 10^{-7}$:

```
>> x=linspace(10^-8,10^-7);
>> f=inline('sqrt(1-cos(x).^2)./sin(x)');
>> plot(x,f(x))
```

Esercizio 2
Il calcolo della funzione

$$f(x) = x\sqrt{x^2+1} - x^2$$

risulta instabile per valori di x abbastanza elevati, a causa della evidente cancellazione presente nella formula che definisce la funazione stessa. Tale cancellazione è facilmente evitabile se si calcola la funzione come

$$f(x) = \frac{x}{\sqrt{x^2+1}+x} \qquad .$$

Si confrontano i grafici ottenuti calcolando la funzione nei due modi proposti, per $10^7 \le x \le 10^8$:

```
>> x=linspace(10^7,10^8);
>> f1=inline('x.*sqrt(x.^2+1)-x.^2');
>> f2=inline('x.*1./(sqrt(x.^2+1)+x)');
>> plot(x,f1(x),x,f2(x))
```

Si osservi che $\lim_{x\to\infty} f(x) = 1/2$.

Esercizio 3
Il calcolo della funzione

$$f(x) = \frac{\log(1+x)}{x}$$

dà risultati poco accurati, per $10^{-15} \le x \le 10^{-14}$

```
>> x=linspace(10^-15,10^-14);
>> f1=inline('log(1+x)./x');
>> plot(x,f1(x))
```

a causa della instabilità del calcolo del logaritmo nell'intorno di 1. Sorprendentemente, se si calcola la funzione come

$$f(x) = \frac{\log(1+x)}{(x+1)-1}$$

si ottengono risultati molto più precisi

```
>> hold
Current plot held
>> f2=inline('log(1+x)./(x+1-1)');
>> plot(x,f2(x))
>> hold
Current plot released
```

se si tiene conto che $\lim_{x\to 0} f(x) = 1$.
La spiegazione, non ovvia, sta nel fatto che l'errore di cancellazione introdotto al denominatore compensa l'errore (anch'esso imputabile a una cancellazione) che si è commesso nel calcolo di $\log(1+x)$.

Esercizio 4 Si consideri l'equazione $f(x) = 0$, con

$$f(x) = x^3 + x - 1 \qquad .$$

Trattandosi di una equazione algebrica, Matlab calcola tutte le soluzioni in campo complesso con la funzione

```
roots
```

che vuole come argomento il vettore dei coefficienti del polinomio $x^3 + x - 1$.

```
>> roots([1 0 1 -1])
```

Vogliamo tuttavia studiare il comportamento di alcuni metodi iterativi.
Separiamo graficamente le soluzioni reali, intersecando i grafici di $y = x^3$ e di $y = -x + 1$:

```
>> x=linspace(-2,2);
>> plot(x,x.^3,x,-x+1,x,zeros(1,100))
```

La separazione indica che esiste una sola soluzione reale α, $0 < \alpha < 1$. Tracciando il grafico di $f(x)$

```
>> fplot('x.^3+x-1',[0 1])
>> hold
Current plot held
>> fplot('0',[0 1])
>> hold
Current plot released
```

si osserva che $f(0)f(1) < 0$ e questo è sufficiente a garantire la convergenza del metodo di bisezione:

```
function [x,fx]=bisezione(f,a,b,tol,n)
d=sign(feval(f,a)); i=0; ind=1;
while ind & i<n
    i=i+1; x0=(a+b)/2; f0=feval(f,x0);
    if d*sign(f0)>0 a=x0; else b=x0; end
    ind= b-a>tol; x(i)=x0; fx(i)=f0;
end
```

```
>> f=inline('x.^3+x-1');
>> [x,fx]=bis(f,0,1,10^-14,1000);
>> length(x)

ans =

    47

>> x(47)

ans =

    0.6823
```

Nei vettori

```
x, fx
```

sono stati salvati rispettivamente le iterate del metodo e i relativi valori della funzione. Sono state effettuate 47 iterazioni.
Dalla separazione grafica risulta anche che, nell'intervallo di separazione, $f'(x) > 0$, $f''(x) > 0$, e $f(1)f''(1) > 0$, quindi si può usare il metodo delle tangenti con $x_0 = 1$, per ottenere una successione convergente monotona:

```
function [x,fx]=tangenti(f,f1,a,b,x0,tol,n)
i=0; ind=1;
f0=feval(f,x0);
while ind & i<n
    i=i+1;
```

```
    x1=x0-f0/feval(f1,x0);
    ind=abs(x1-x0)>tol;
    f0=feval(f,x1);
    x(i)=x1;
    fx(i)=f0;
    x0=x1;
end

>> f1=inline('3*x.^2+1');
>> [x,fx]=tangenti(f,f1,0,1,1,10^-14,10);
>> x

x =

    0.7500    0.6860    0.6823    0.6823    0.6823    0.6823

>> fx

fx =

    0.1719    0.0089    0.0000    0.0000         0         0
```

che mostra come siano state sufficienti 6 iterazioni, invece delle 47 effettuate con il metodo di bisezione.