Notas de la clase de 6 abril, 2016

- Acerca del prob. 2a de la tarea 8: la fórmula correcta para el polinomio de Taylor de grado n, donde $n = 1, 3, 5, \ldots$ (un entero positivo impar), es

  $$p_n(x) = \sum_{i=0}^{(n-1)/2} (-1)^i \frac{x^{2i+1}}{(2i+1)!}.$$

  (tenía error dos veces en esta fórmula, pero creo que ya es correcta).

  Para convertir esta fórmula en código, uno puede casi copiar la fórmula

  ```
  function y=p(n,x)
  y=0;
    for i=0:(n-1)/2
    y=y+(-1)^i*x.^(2*i+1)/factorial(2*i+1);
  end
  ```

  Nota que esta función está vectorizada en x, pero no en n. Así que, por ejemplo, si queremos graficar varios de los $p(n, x)$, tenemos que usar un ciclo.

  ```
  x=-5:.1:5;
  hold on
  for n=1:2:11
    plot(x,p(n,x))
  end
  plot(x,sin(x))
  plot([-5 5], [0 0],[0 0], [-2 2])  % ejes de coordendas
  ylim([-2 2])
  ```

- Algorítmo de ordenamiento por inserción (prob. 4 de la tarea 8). Hemos redactado en la clase el siguiente algoritmo

  ```
  function B=ord_ins(A)
  B=A;
  for i=2:length(A)
      j=i-1;
      while B(j)>A(i)
          j=j-1;
          if j==0 break
          end
      end
      if j+1<i
          for k=i:-1:j+2
              B(k)=B(k-1);
          end;
          B(j+1)=A(i);
      end
  end
  end
  ```

- Aquí esta una versión más vectorizada, con un solo ciclo, más breve y facil de leer:

```
function B=ord_ins(A)
B=A;
for i=2:length(A)
    j=sum(B(1:i-1)<A(i));
    if j+1<i
        B(j+2:i)=B(j+1:i-1);
        B(j+1)=A(i);
    end
end
end
```

- El método de Newton. Para encontrar soluciones de una ecuación $f(x) = 0$, uno empieza con una "adivinanza" inicial x_1, luego define de manera sucesiva $x_2, x_3, x_4, \ldots$ por la fórmula

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Por ejemplo, usamos este método para encontrar la $\sqrt{2}$. O sea, la solución positiva de $f(x) = x^2 - 2 = 0$. Aquí la fórmula general nos da (después de simplificarla)

$$x_{n+1} = x_n/2 + 1/x_n.$$

Una manera de asegurar que estamos cerca a una solución de $f(x) = 0$ es pedir que $|f(x)| < E$, para un E pequeño, digamos $E = 10^{-8}$.

```
function [x i]=raiz2[E]
   x=2; i=0;
   while abs(x^2-2)>E
      x=x/2 + 1/x;
      i=i+1;
   end
end

>> format long

>> [x  i]=raiz2[10^-8]
x =
   1.414213562374690
i =
    4

>> sqrt(2)
ans =
   1.414213562373095
```

OJO: El método no siempre funciona. Por ejemplo, necesitas que $f'(x_n) \neq 0$. No entramos en los detalles en este curso, pero pueden suceder cosas bien complicadas, dependiendo de la f. Pero cuando funciona, es muy eficiente (converge muy rápidamente).