- La duración del examen será de 1h y 40 minutos.
- La nota del examen será la media aritmética de los problemas propuestos.
- Las notas se publicarán en Moodle. Fecha prevista de publicación de notas: 6/7/17. Fecha prevista de revisión presencial del examen: 7/7/17.

Problema 1.

Se considera una representación en coma flotante en base 2. Cada palabra utiliza en memoria los siguientes 6 bits: 3 bits para el exponente, $e=(e_1\ e_2\ e_3)$, y 3 bits para la mantisa, $m=(b_1\ b_2\ b_3)$. Los números máquina $\hat{x}$ representados y su denominación son los siguientes:

Si $e=(0\,0\,0)_2=0$, $\hat{x}=(0.b_1b_2b_3)_2\times 2^{-3}$ Número denormalizado

Si $e=(e_1\ e_2\ e_3)_2\neq 0$, $\hat{x}=(1.b_1b_2b_3)_2\times 2^{e-4}$ Número normalizado

En esta representación:

- ¿Cuántos números denormalizados hay?, ¿cuántos normalizados? y, ¿cuántos números máquina?
- Calcular los números máquina (en formato decimal) cuando se almacena en memoria el contenido de la siguiente tabla. Completar la tabla, expresando el número en formato decimal (p.e., 0.00123, 2.45, etc.).

$e_1\ e_2\ e_3$	$b_1\ b_2\ b_3$	Nº máquina
000	000	
000	001	
000	111	
001	000	
111	111	
100	000	
111	001	

- A partir de los datos de la tabla anterior, calcular el $\text{eps}\left(\frac{7.5}{64}\right)$.
- Calcular los números máquina (en formato decimal) y los valores a almacenar en memoria de los valores que se indican. Completar la siguiente tabla.

	Nº máquina	$e_1\ e_2\ e_3$	$b_1\ b_2\ b_3$
0.1			
$1+0.1$			

Nota: No debéis entregar el enunciado. Copiar las dos tablas en la hoja de respuestas y completarlas. Debéis entregar también vuestro desarrollo del ejercicio, cálculos, etc.

Problema 2.

2.1. - Dar la expresión mediante la fórmula de Newton generalizada de un polinomio p(x) de grado dos que interpola los datos: p(0)=0, p'(0)=1 y p(1)=10.

- Se considera la función a trozos s(x)

$$s(x)=\begin{cases} ax+x^2 & x\in[-1,0] \\ p(x) & x\in[0,1] \end{cases}$$

donde p(x) es el polinomio calculado anteriormente. Determinar el valor del parámetro a para s(x) sea una función spline cuadrado.

2.2. - Se considera el problema de ajustar los datos de la siguiente tabla

x_i	-2	-1	0	1	2
y_i	1	3	8	3	1

por una función aproximante del tipo $u(x)=\dfrac{a}{1+bx^2}$. Transformar dicho problema en un problema de ajuste lineal y construir el sistema lineal sobredeterminado Hc=B (siendo c el vector de incógnitas) al que se llega. Se pide: Dar las expresiones de la matriz H de coeficientes y del término independiente B del sistema.

- Si se ajustaran los datos por una función del tipo $u(x)=1/(1+bx^2)$ ¿el error del ajuste sería mayor o menor que el producido en el caso anterior? Dar una respuesta razonada, sin calcular explícitamente los errores referidos.

Problema 3.

Se quiere aplicar el método iterativo de Newton para resolver la ecuación:

$$\log(x)=4-x$$

3.1. Demostrar que dicha ecuación tiene una raíz s en el intervalo [2,4].

3.2. ¿Qué valores iniciales $x_0 \in [2,4]$ hacen que el método sea convergente?

3.3. Dar la ecuación que implementa el método de Newton para aproximar la raíz s y calcular la primera iteración x_1 partiendo del valor $x_0=4$.

3.4. Si x_0 es cualquier valor del intervalo [2,4] ¿Cuántas iteraciones hay que realizar para aproximar la raíz s con 10 decimales exactos?

Nota: Si $g(x)=\log(x)$; $g'(x)=1/x$; $g''(x)=-1/x^2$ (log(x) denota el logaritmo neperiano de x).

SOLUCIONES

Problema 1.

- En esta representación hay 8 números denormalizados, 56 números normalizados y 64 números máquina.
-

$e_1\ e_2\ e_3$	$b_1\ b_2\ b_3$	Nº máquina
000	000	0
000	001	$2^{-6}=1/64=0{,}015625$
000	111	$7/64=0{,}109375$
001	000	$1/8=0{,}125$
111	111	15
100	000	1
111	001	9

- El número real $\frac{7.5}{64}$ se encuentra entre los siguientes números máquina consecutivos

$$\frac{7}{64}<\frac{7.5}{64}<\frac{8}{64}=\frac{1}{8}$$

Por tanto, $\operatorname{eps}\left(\frac{7.5}{64}\right)=\frac{8}{64}-\frac{7}{64}=\frac{1}{64}$.

-

	Nº máquina	$e_1\ e_2\ e_3$	$b_1\ b_2\ b_3$
0.1	$6/64=0{,}09375=(0.110)_2\times 2^{-3}$	000	110
$1+0.1$	$1=(1.000)_2\times 2^{0}$	100	000

Problema 2.

2.1. - La función pedida es un polinomio p(x) de grado dos, que tiene que verificar p(0)=0, p'(0)=1 y p(1)=10. Para dar la Fórmula de Newton generalizada de dicho polinomio, primero construimos la tabla de diferencias divididas generalizada con los datos dados:

$$\begin{array}{cccc}
x_k & p(x_k) & p[.,.] & p[.,.,.] \\
x_0=0 & 0 & p[0,0]=p'(0)=1 & p[0,0,1]=9 \\
x_0=0 & 0 & p[0,1]=10 & \\
x_1=1 & 10 & &
\end{array}$$

Por tanto, la Fórmula Newton generalizada del polinomio pedido es:

$$\boxed{p(x)=p[0]+p[0,0]x+p[0,0,1]x^2=x+9x^2.}$$

- Se considera la función a trozos s(x)

$$s(x)=\begin{cases} ax+x^2 & x\in[-1,0] \\ p(x) & x\in[0,1] \end{cases}$$

donde p(x) es el polinomio calculado anteriormente. Para que la función s(x) sea una función spline cuadrática se tiene que verificar:

1. s(x) sea un polinomio a trozos de grado 2 en los intervalos considerado: Esto se verifica para cualquier valor de a.
2. s(x) tiene que ser continua en el intervalo (-1,1): Esto se verifica para cualquier valor de a.
3. s'(x) tiene que ser continua en el intervalo (-1,1). Para que esto se verifique:
   $s'(0^-)= s'(0^+)$ <->a=p'(0)=1. Por tanto a=1.

2.2. - Se considera el problema de ajustar los datos de la tabla por una función aproximante del tipo $u(x)=\dfrac{a}{1+bx^2}$.

Evaluando el tipo de función pedido en los datos dados se llega al sistema (no lineal)

$$u(x_i)=\frac{a}{1+bx_i^{\,2}}=y_i,\quad i=1,\ldots,5.$$

Un posible forma de transformar el sistema anterior en un sistema lineal (multiplicando por denominador):

$$a=(1+bx_i^{\,2})y_i,\quad i=1,\ldots,5\rightarrow \boxed{a-x_i^{\,2}y_ib=y_i,\quad i=1,\ldots,5}.$$

Efectuando las operaciones y escribiendo el sistema en forma matricial Hc=B, resulta:

$$a-\underbrace{x_i^{\,2}y_i}b\ =y_i\quad i=1,\ldots,5$$

$$\downarrow\quad\downarrow\qquad\downarrow$$

$$\underbrace{\begin{pmatrix}1&-4\\1&-3\\1&0\\1&-4\\1&-3\end{pmatrix}}_{H}\underbrace{\begin{pmatrix}a\\b\end{pmatrix}}_{c}=\underbrace{\begin{pmatrix}1\\3\\8\\3\\1\end{pmatrix}}_{B}$$

- La función $u(x)=1/(1+bx^2)$ es un subtipo de la función anterior (en la que el parámetro a ya tiene un valor determinado). Por tanto el error del ajuste en este caso será mayor o igual que el producido en el caso anterior.

Problema 3.

a) La ecuación $log(x)=4-x$, se puede poner en la forma $f(x)=0$ haciendo

   $f(x)=log(x)-4+x$

   La función $f(x)$ anterior cumple las condiciones:

   - Es una función continua en el intervalo $[2,4]$
   - $f(2)f(4)<0$ Cambia de signo en los extremos del intervalo $[2,4]$

   El teorema de Bolzano asegura que si se cumplen las condiciones anteriores, existe al menos un $s\in(2,4)$ tal que $f(s)=0$, es decir, existe al menos una raíz de la ecuación $f(x)=0$ en el intervalo $[2,4]$

b) El método iterativo de Newton para resolver ecuaciones, tiene la expresión general

$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$; $n = 0, 1, 2, \ldots$ y converge cuando se verifica

$Me_0 < 1$ siendo

$M = \frac{max|f''(x)|}{2\,min|f'(x)|}$; $x \in [a, b]$ y $e_0 = |x_0 - s|$

En nuestro caso

$f'(x) = \frac{1}{x} + 1$; $f''(x) = -\frac{1}{x^2}$

En el intervalo $[2{,}4]$ se cumple:

- $|f''(x)| = \frac{1}{x^2}$ función continua y decreciente en $[2{,}4]$ por tanto,

  $max|f''(x)| = \frac{1}{2^2}$
- $|f'(x)| = \frac{1}{x} + 1$ que es también continua y decreciente en $[2{,}4]$ por tanto,

  $min|f'(x)| = \frac{1}{4} + 1$

Sustituyendo $M = \frac{1}{10}$

En este caso, el método de Newton converge siempre que $\frac{1}{10} e_0 < 1;\ es\ decir, e_0 < 10$

Como $x_0 \in [2{,}4]$ y $s \in (2{,}4)$ siempre se cumple que $e_0 = |x_0 - s| < 2$ y el método, por tanto, converge para cualquier $x_0 \in [2{,}4]$

c) La ecuación que implementa el método de Newton para resolver nuestra ecuación es:

$$x_{k+1} = x_k - \frac{log(x_k) - 4 + x_k}{\frac{1}{x_k} + 1}; k = 0, 1, 2, \ldots$$

Si el valor inicial x_0=4, sustituyendo, tenemos $x_1 = 2{,}8910$

d) Si $x_0 \in [2{,}4]$ el error inicial $e_0 = |x_0 - s| < 2$

Llamando $e_k = |x_k - s|$ al error cometido en la k-ésima iteración se puede demostrar que:

$$e_k \leq \frac{1}{M}(Me_0)^{2^k}$$

Para $M = \frac{1}{10}$, y garantizar al menos 10 decimales exactos, se debe cumplir

$$e_k \leq \frac{1}{1/10}\left(\frac{1}{10}2\right)^{2^k} < 10^{-10}$$

Operando

$2^k > 12{,}876 \ldots que\ se\ cumple\ para\ k \geq 4$

Hay que hacer como mínimo 4 iteraciones para garantizar al menos 10 decimales exactos en el resultado.

ALGORÍTMICA NUMÉRICA **30 Junio 2017**

Examen Computacional

Ejercicio 1.

Consideramos las dos siguientes expresiones que dan resultados similares para valores de h cercanos a cero:

$$\cosh(1) \cong \frac{\cosh(1+h) - 2\cosh(1) + \cosh(1-h)}{h^2}$$

Se pide:

- Construir un vector n con valores 1, 2, 3,…, 8.
- A partir de n construir un vector h con valores 0.1, 0.01, 0.001,….., 0.00000001.
- Construir un vector con los resultados de evaluar la expresión de la derecha para el vector h.
- Calcular el error relativo de los resultados de la expresión de la derecha con respecto a la de la izquierda.
- Dibujar, en la escala adecuada, la relación entre n y el error relativo (los valores de n deben ir en el eje horizontal y los del error relativo en el vertical).
- Calcular el número de cifras decimales correctas entre ambas expresiones.
- Dibujar, en la escala adecuada, la relación entre h y las cifras (los valores de h deben ir en el eje horizontal y los de las cifras en el vertical).
- Para que valor de h se consiguen 8 cifras correctas.

Ejercicio 2.

Ejecutar los comandos xi=pi*[-1.5:1:1.5]',yi=sin(xi). Sea la tabla de datos $\{\mathrm{xi}, \mathrm{yi}\}$.

a) Interpolar la tabla de datos mediante un polinomio de grado mínimo.

Dibujar la gráfica del polinomio de interpolación en el intervalo [-5, 5] ('b.' en azul) junto con los valores de la tabla ('ro' en rojo).

b) Interpolar la misma tabla de datos con un polinomio que verifique la condición $p(0)=1$.
Dibujar la gráfica del polinomio en el intervalo [-5, 5] ('g.' en verde), con los valores de la tabla (en rojo), junto con el punto $(0,1)$ ('sr' cuadrado rojo).

c) Ajustar los datos de la tabla con un polinomio de grado 3 que verifique la condición $p'(0)=1$.
Dibujar la gráfica del polinomio de ajuste en el intervalo [-5, 5] ('k.' en negro), con los valores de la tabla (en rojo). Calcular el vector residuo y el error del ajuste.

Dibujar en una misma gráfica los tres polinomios obtenidos en los apartados anteriores, junto con los valores de la tabla y el punto $(0,1)$.

Ejercicio 3.

Dada la función $f(x) = x^2 - e^{-x}$.

a) Realizar una gráfica de la función f(x) en el intervalo [-2, 2]. Dar un intervalo aproximado con una longitud máxima de 1 donde se encuentre la raíz a partir de la gráfica. Demostrar analíticamente que en dicho intervalo existe al menos una raíz.

b) Implementar y ejecutar el siguiente método $x_{n+1} = \sqrt{e^{-x_n}}$ para encontrar la raíz de f(x) partiendo de $x_0 = 1$
El método deberá iterar hasta que el error $e_n \approx |x_n - x_{n-1}| < 1e-10$. El código deberá en cada iteración imprimir, el número de iteración, la raíz obtenida y el error utilizando el siguiente formato:

'Iter %d Sol %.15f Error %e\n'

c) Implementar y ejecutar el siguiente método

$$z_0 = a, z_{n+1} = z_n - \frac{f(z_n)}{f'(x_n)}$$
$$x_0 = b, x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

que calculará la raíz x_n de f(x) partiendo de $a = 0.5, b = 1.0$ e iterando hasta que el error $e_n \approx |x_n - z_n| < 1e-10$. El código deberá en cada iteración imprimir, el número de iteración, la raíz obtenida y el error utilizando el siguiente formato:

'Iter %d Sol %.15f Error %e\n'

d) A la vista de los resultados obtenidos con ambos métodos: ¿Cuál es el orden de convergencia de cada método? Justificar la respuesta.

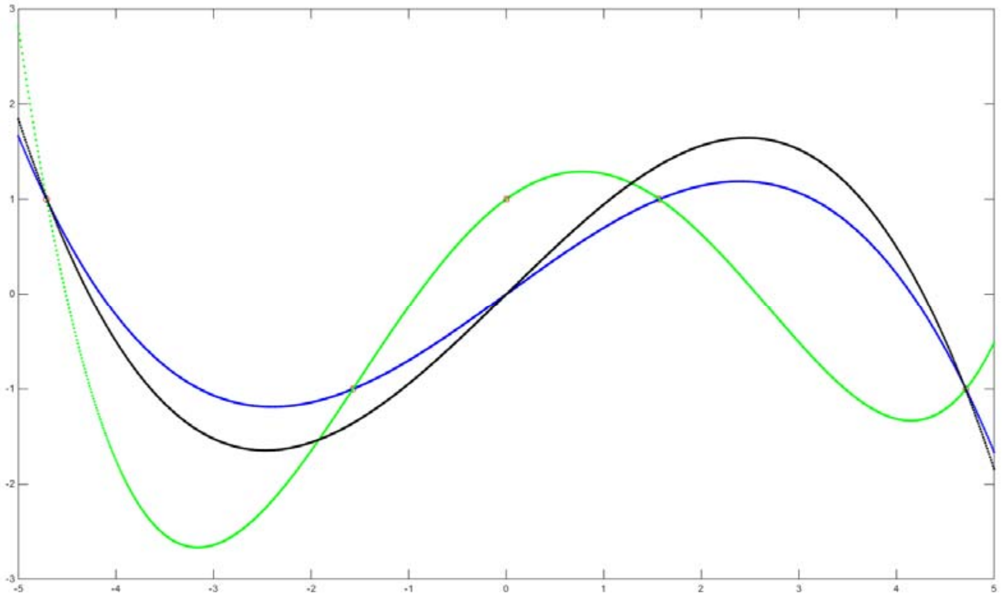
<u>Solución</u>

Ejercicio 1

```
h=10.^-[1:8]
n=[1:8]
h=10.^-n
der = (cosh(1+h)-2*cosh(1)+cosh(1-h))./(h.^2)
izq = cosh(1)
er=abs(der-izq)/abs(izq)
semilogy(n,er)
figure(1),semilogy(n,er)
cif = -log10(er)
figure(2),semilogx(h,cif)
% No se consiguen 8 cifras. Lo más cercano (casi 8) es para h = 10^ -4 que es
0.0001
```

Ejercicio 2.

```
clear
xi=pi*[-1.5:1:1.5]',yi=sin(xi)

a)
H=[xi.^0 xi.^1 xi.^2 xi.^3];b=yi;c=H\b;
xx=-5:0.01:5;px=c(1)+c(2)*xx+c(3)*xx.^2+c(4)*xx.^3;
plot(xi,yi,'ro',xx,px,'b.')

b) Opción 1

xi(5)=0,yi(5)=1,
H21=[xi.^0 xi.^1 xi.^2 xi.^3 xi.^4];b21=yi;c21=H21\b21;
px21=c21(1)+c21(2)*xx+c21(3)*xx.^2+c21(4)*xx.^3+c21(5)*xx.^4;
plot(xi,yi,'ro',xx,px,'b.',xx,px21,'g.',0,1, 'rs')

b) Opción 2
% p(0)=1, a=1, px=1+bx+c x^2+d x^3+e x^4;
H2=[xi.^1 xi.^2 xi.^3 xi.^4];b2=yi-xi.^0;c2=H2\b2;
px2=1+c2(1)*xx+c2(2)*xx.^2+c2(3)*xx.^3+c2(4)*xx.^4;
plot(xi,yi,'ro',xx,px,'b.',xx,px2,'g.',0,1, 'rs')

c)
% p’(0)=1, b=1, px=a+x+c x^2+d x^3;
H3=[xi.^0 xi.^2 xi.^3];b3=yi-xi;c3=H3\b3;
px3=c3(1)+xx+c3(2)*xx.^2+c3(3)*xx.^3;
plot(xi,yi,'ro',xx,px,'b.',xx,px2,'g.',0,1, 'rs',xx,px3,'k.')
r=c3(1)+xi+c3(2)*xi.^2+c3(3)*xi.^3-yi
E=sum(r.^2)
r =
    0.0133
   -0.3587
    0.3587
   -0.0133

E =
    0.2577
```



Ejercicio 3.

```
clc
clear

x=(-2:0.01:2);
fx=x.^2-exp(-x);
plot(x,fx,x,fx*0)
f0=0.^2-exp(-0);
f1=1.^2-exp(-1);
f0*f1

error=100.0;
iteracion=0;
xn=1.0;
while (error>1e-10)
    iteracion=iteracion+1;
    xn1=sqrt(exp(-xn));
    error=abs(xn1-xn);
    fprintf('Iter %d Sol %.15f Error %e\n', iteracion,xn1,error)
    xn=xn1;

end

fprintf('\n \n')
z=0.5;
x=1.0;
iteracion=0;
error=100.0;
while(error>1e-10)
    iteracion=iteracion+1;
    z=z-((z^2-exp(-z)) / (2*x+exp(-x)));
    x=x-((x^2-exp(-x)) / (2*x+exp(-x)));
    error=abs(z-x);
```

```
    fprintf('Iter %d Sol %.15f Error %e\n', iteracion,x,error)

end
```

```
Iter 1 Sol 0.606530659712633 Error 3.934693e-001
Iter 2 Sol 0.738403149974731 Error 1.318725e-001
Iter 3 Sol 0.691286050428152 Error 4.711710e-002
Iter 4 Sol 0.707765096310001 Error 1.647905e-002
Iter 5 Sol 0.701957408706619 Error 5.807688e-003
Iter 6 Sol 0.703998745804550 Error 2.041337e-003
Iter 7 Sol 0.703280563001844 Error 7.181828e-004
Iter 8 Sol 0.703533150353015 Error 2.525874e-004
Iter 9 Sol 0.703444304176034 Error 8.884618e-005
Iter 10 Sol 0.703475554038709 Error 3.124986e-005
Iter 11 Sol 0.703464562367352 Error 1.099167e-005
Iter 12 Sol 0.703468428503616 Error 3.866136e-006
Iter 13 Sol 0.703467068652530 Error 1.359851e-006
Iter 14 Sol 0.703467546957921 Error 4.783054e-007
Iter 15 Sol 0.703467378721781 Error 1.682361e-007
Iter 16 Sol 0.703467437896102 Error 5.917432e-008
Iter 17 Sol 0.703467417082498 Error 2.081360e-008
Iter 18 Sol 0.703467424403344 Error 7.320846e-009
Iter 19 Sol 0.703467421828356 Error 2.574988e-009
Iter 20 Sol 0.703467422734066 Error 9.057102e-010
Iter 21 Sol 0.703467422415497 Error 3.185688e-010
Iter 22 Sol 0.703467422527548 Error 1.120514e-010
Iter 23 Sol 0.703467422488136 Error 3.941225e-011
```

```
Iter 1 Sol 0.733043605245445 Error 8.247401e-002
Iter 2 Sol 0.703807786324133 Error 2.631407e-003
Iter 3 Sol 0.703467468331798 Error 2.738812e-006
Iter 4 Sol 0.703467422498392 Error 2.968181e-012
```