INFORMACIÓN BÁSICA						
Nombre del Curso	**Fecha de diligenciamiento**(dd/mm/aaaa)		**Sección(es)**		**Periodo académico**	
Computación Científica en IEE	**01/02/2015**		**1-2**		**201610**	
Nombre de la práctica:	**Sistemas Numéricos de Punto Flotante**			**Práctica No.:**	**1**	
Profesor(es):	**Nestor Peña Traslaviña**		**Asistente(es) Graduado(s):**	**Daniel Felipe Duarte Sánchez**		
Semana de la práctica (1-16)	**Versión de la guía**		**Nomenclatura del espacio a utilizar**			
3	**1.0**		**ML-108A / ML-107**			

CONTENIDO DE LA GUÍA

Objetivos

- Entender las características, ventajas y desventajas asociadas con las unidades de punto flotante que permiten el desarrollo de la computación numérica.
- Observar casos que ilustran las precauciones que deben ser tenidas en cuenta en sistemas intensivos en computación numérica.

Procedimiento de la práctica de laboratorio

Epsilon de una máquina (ϵ_m)

El valor épsilon de una máquina es la diferencia entre 1.0 y el número de punto flotante siguiente. Ya que este espaciamiento no es uniforme en todo el rango de representación de números de punto flotante, esta definición puede ser extendida alrededor de cualquier otro número (diferente de 1.0) y escrita como $\epsilon(x)$ para cualquier número de punto flotante x. Esta es una de las razones para calcular el valor de épsilon en Matlab (el cual cuenta por defecto con un valor constante "eps"), adicionalmente, es conveniente conocer algunos algoritmos para computar este valor, ya que algunos lenguajes de programación no cuentan un valor predefinido del mismo.

Ejercicios:

1. La función MaqEps aprovecha el hecho de que $fl(1.0 + \epsilon_m) > 1.0$. Inicialmente, esta función inicializa épsilon = 1.0, luego lo divide repetidamente entre 2 hasta que se cumpla $fl(1.0 + \epsilon_m) \leq 1.0$. Tenga en cuenta que en cada iteración $fl(1.0 + \epsilon_m) = 1.0 + \epsilon_m$, excepto al final de la última iteración. Es decir, genera una secuencia de números de punto flotante a excepción del último. Esto debe ser tenido en cuenta para reportar correctamente el valor de ϵ_m. Escriba la función MaqEps en Matlab.
2. Modifique la función MaqEps en Matlab de tal forma que compute $\epsilon(x)$ cualquier número de punto flotante x. Analice el comportamiento de $\epsilon(x)$ como función de x. Se recomienda hacer una gráfica x vs. $\epsilon(x)$.
3. Consulte el procedimiento Kahan [1] cuya referencia encuentra en SicuaPlus, para calcular ϵ_m, y repita los puntos 1 y 2 haciendo uso de este algoritmo. Compare los resultados obtenidos en ambos algoritmos. Observe que dependiendo de la forma en que implemente el algoritmo de Kahan para $\epsilon(x)$, es posible que para los múltiplos de 3 la función entregue: $\epsilon(x) = 0$. En este caso estime $\epsilon(x) \approx \epsilon(x + 1)$.

Números de punto flotante

4. Investigue el estándar *IEEE 754-2008*. Luego encuentre en Matlab los valores constantes: eps, realmax, realmin, 1/realmin y 1/realmax. Justifique estos valores en términos del estándar IEEE.
5. Determine la respuesta de Matlab a los valores indeterminados:

$$\frac{0}{0}, \frac{\infty}{\infty}, \infty * 0, 1^{\infty}, \infty - \infty, 0^{0}, \infty^{0}$$

   Está de acuerdo con los resultados mostrados por Matlab? (Conviene consultar las formas indeterminadas) Explique.

6. Considere que analíticamente: $\sin(\pi) = 0,\ \cos(\pi) = -1,\ \sin(x)^2 + \cos(x)^2 = 1$. Qué valores entrega Matlab para sin(pi), cos(pi), sin(pi)ˆ2+cos(pi)ˆ2? Explique detalladamente.
7. Explique los resultados de las siguientes operaciones teniendo en cuenta que Matlab hace uso del estándar IEEE definido en el punto 4.

```
>> (1 + 1e-16) - 1
ans = 0
>> (1 + 2e-16) - 1
ans = 2.2204e-16
>> (1 - 1e-16) - 1
ans = -1.1102e-16
>> 1 + (1e-16 - 1)
ans = 1.1102e-16
```

8. Evalúe $\sum_{k=1}^{3000} \frac{1}{k}^{2} = 1.6446$ redondeando todos los resultados intermedios a 4 cifras significativas.

```
>> sum = 0;
>> for k=1:3000
sum = chop(sum+1/k^2, 4);
end
>> sum
sum = 1.6240
```

   Este resultado es correcto solo hasta su segundo dígito, sin embargo no hay cancelación (no hay restas). Explique y proponga un mejor método.

9. El número $e = 2.7182818 \ldots$ puede ser definido como:

$$e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^{n}$$

   Esto sugiere que un algoritmo para calcular e consiste en escoger n lo suficientemente grande, tal que:

$$\hat{e} = \left(1 + \frac{1}{n}\right)^{n}$$

Diseñe un script que encuentre en tiempo razonable (tenga que buscar entre valores consecutivos de n es muy ineficiente) un valor óptimo aproximado de n. Explique los resultados encontrados.

Entrega

Para este laboratorio debe entregarse un archivo zip que contenga un informe (de acuerdo a las reglas del curso) y cada uno de los scripts o funciones debidamente documentados. Para el informe tenga en cuenta las pautas dadas por el asistente graduado.

La entrega debe hacerse por SicuaPlus con un plazo máximo de 10 minutos antes de la práctica de laboratorio.

Bibliografía recomendada

[1] D. O´Connor. Lab Exercise No. 3: MATLAB´s Floating Point Number System. Numerical Algorithms. 2006

[2] Shinnerl. Floating-Point Representation and Approximation, Errors, cancellation, instability, simple one-variable examples, swamping. EE103.

[3] R. Burden and J.D. Faires. Numerical Analysis, Ninth Edition. BROOKS/COLE ENGAGE Learning.