AMERICAN UNIVERSITY OF BEIRUT
Mathematics Department-FAS

MATH 251
TEST 1
SPRING 2008-2009
Closed Book, 75 mn

STUDENT NAME	
ID NUMBER	

Problem	**Out of**	**Grade**
1	**13**	
2	**12**	
3	**14**	
4	**11**	
TOTAL	**50**	

1. Determine the decimal number x in $F(10, 5, -60, +60)$, that has the following hexadecimal representation in the IEEE single precision system. (Round to the closest, if needed.)

$$x = [7D187ABC]_{16}$$

2. (i) - What are the numbers in the IEEE single precision system F_S, immediately to the right and to the left of $x = 2^m$. (i.e succ(x) and pre(x)). How far is each from 2^m ?
Discuss the case $m = -126$, identify pre(x) in that case, then write the Binary string representation of pre(x) in F_S.

    - m $= -126$

- m = -126

  - $pre(x) =$

  - Binary Bit-string representation of pre(x) :

(ii) - Determine the Cardinality of F_S.

3. Prove that the function $f(x) = ln(1-x) - e^x$, defined for $x < 1$, has a unique negative root, then locate the root:
(a) - By plotting both functions $g(x) = ln(1-x)$ and $h(x) = e^x$.

(b) By studying the behaviour of $f(x)$ and $f'(x)$ on $(-\infty, 1)$

(ii) - How many steps of the bisection method are approximately needed to provide an approximation to a root r , up to 4 decimal figures ?

(iii) - Apply 2 iterations of the Bisection method, to approximate the root of $f(x)$. Give the results in $F(10, 4, -20, +20)$, rounding to the closest if necessary.

$r_1 =$

$r_2 =$

(iv) - Write Newton's iteration formula, then find one iteration in $F(10, 4, -20, +20)$, rounding to the closest if necessary.

$r_{n+1} =$

$r_3 =$

(v) - Write the Secant method iteration formula, then find one iteration in $F(10, 4, -20, +20)$, rounding to the closest if necessary.

$r_{n+1} =$

$r_3 =$

4. Consider the following $n * n$ **quadridiagonal matrix**

$$A = \begin{pmatrix} d_1 & u_1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ l_1 & d_2 & u_2 & 0 & \cdots & \cdots & 0 & 0 \\ v_1 & l_2 & d_3 & u_3 & 0 & \cdots & 0 & 0 \\ 0 & v_2 & l_3 & d_4 & u_4 & 0 & \cdots & 0 \\ 0 & 0 & v_3 & l_4 & d_5 & u_5 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & v_{n-3} & l_{n-2} & d_{n-1} & u_{n-1} \\ 0 & 0 & \cdots & 0 & 0 & v_{n-2} & l_{n-1} & d_n \end{pmatrix}$$

This matrix is assumed to have the diagonal dominance property: As such we can use naive Gauss elimination to reduce it into an upper triangular matrix.

(i) - Complete the following algorithm that should perform the reduction of the matrix A:

```
function [d,u,l,v]=lu_QuadriDiag(d,u,l,v)
% d is the diagonal vector. Its dimension is n
% u is the upper diagonal vector. Its dimension is n-1
% l is the first lower diagonal vector. Its dimension is n-1
% v is the second lower diagonal vector. Its dimension is n-2

for k = 1 : n-1
% Get the multipliers (take into consideration the last iteration)
      ..................................................................
      if ...............................................................
              ..........................................................
     end

% Modify the coefficients (take into consideration the last iteration)
      ..................................................................
      if ...............................................................
              ..........................................................
     end
 end
```

(ii) - Find the number of flops needed to execute the above algorithm.

(iii) - Give the Upper Triangular matrix U obtained from this reduction.