**University of May 8, 1945 - Guelma**
**Faculty of Mathematics and Computer Science**
**Department of Computer Science**
**2nd Year Bachelor's Degree - Computer Science**
**2024/2025**
**Dr. Chemseddine Chohra**

# Exam: Numerical Methods - Correction

## Questions: (5 pts)

Choose the correct answer (only one):

1. The Jacobi method is:
   A. A direct method for solving linear systems.
   B. An iterative method for solving linear systems. **(Correct Answer)** **(1 pt)**
   C. A method for calculating the determinant of a matrix.
2. The eigenvectors of a matrix A associated with an eigenvalue $\lambda$ are obtained by solving the equation:
   A. $(A - \lambda I)x = 0$ **(Correct Answer)** **(1 pt)**
   B. $(A + \lambda I)x = 0$
   C. $(A - \lambda I)x = b$
3. The convergence of the Jacobi method is guaranteed if:
   A. The matrix A is strictly diagonally dominant. **(Correct Answer)** **(1 pt)**
   B. The matrix A is symmetric.
   C. The matrix A is triangular.
4. In the IEEE-754 format, double precision (64 bits) uses:
   A. 11 bits for the exponent and 52 bits for the mantissa. **(Correct Answer)** **(1 pt)**
   B. 8 bits for the exponent and 23 bits for the mantissa.
   C. 10 bits for the exponent and 21 bits for the mantissa.
5. A vector norm is zero if and only if:
   A. Only one element of the vector is zero.
   B. All elements of the vector are zero. **(Correct Answer)** **(1 pt)**
   C. The sum of the elements of the vector is zero.

## Exercise 1: (4 pts)

In this exercise, we will use the **binary8** format to encode (and decode) floating-point numbers. This format uses:

- 1 sign bit,
- 4 bits for the exponent,
- 3 bits for the mantissa (+1 normalization bit),
- with an exponent bias X = 7.

1. **Encode** the following decimal numbers in the **binary8** format (round to the nearest):
   - A = 12.75
     **Solution:**
     A = 12.75 in binary is 1100.11. Normalized: $1.10011 \times 2^3$.
     When rounded to 3 bits: $1.101 \times 2^3$.
     Sign bit: 0 (positive)
     Exponent: $3 + 7 = 10$ (in binary: 1010)
     Mantissa: 101
     **Encoded:** $(01010101)_{b8}$ **(0.75 pts)**

- B = -1.25
  **Solution:**
  B = -1.25 in binary is $-1.01$. Normalized: $-1.01 \times 2^0$.
  Sign bit: 1 (negative)
  Exponent: $0 + 7 = 7$ (in binary: 0111)
  Mantissa: 010
  **Encoded:** $(10111010)_{b8}$ **(0.75 pts)**

2. **Decode** the following numbers encoded in the **binary8** format (give the result in decimal):
   - C = $(01101011)_{b8}$
     **Solution:**
     Sign bit: 0 (positive)
     Exponent: 1101 (binary) = 13 (decimal)
     Mantissa: 011
     Value: $1.011 \times 2^{13-7} = 1.011 \times 2^6 = (1011000)_{b8} = 88$ (decimal) **(0.5 pts)**
   - D = $(11010101)_{b8}$
     **Solution:**
     Sign bit: 1 (negative)
     Exponent: 1010 (binary) = 10 (decimal)
     Mantissa: 101
     Value: $-1.101 \times 2^{10-7} = -1.101 \times 2^3 = -(1101)_{b8} =$ -13 (decimal) **(0.5 pts)**

3. **Perform** the following operations in the **binary8** format (round to the nearest):
   - A $\otimes$ B
     **Solution:**
     A = 12.75 = 1.101 $\times 2^3$ (in binary scientific notation)
     B = -1.25 = -1.01 $\times 2^0$
     For multiplication, we multiply the mantissas and add the exponents:
     $A \otimes B = 1.101 \times 2^3 \times -1.01 \times 2^0 = -10.00001 \times 2^3$
     When normalized: $-1.000001 \times 2^4$
     When rounded to 3 bits: $-1.000 \times 2^4$
     In decimal: -1.000 $\times 2^4$ = $(-10000)_2$ = -16 **(0.75 pts)**
   - C $\oplus$ D
     **Solution:**
     C = 88 = 1.011 $\times 2^6$
     D = -13 = -1.101 $\times 2^3$
     For addition, we align the exponents and then add the mantissas:
     $C \oplus D = 1.011 \times 2^6 + -1.101 \times 2^3 = 1.011 \times 2^6 + -0.001101 \times 2^6 = 1.001011 \times 2^6$
     When rounded to 3 bits: $1.001 \times 2^6$
     In decimal: 1.001 $\times 2^6$ = $(1001000)_2$ = 72 **(0.75 pts)**

## Exercise 2: (6 pts)

Consider the following linear system written in equation form:

$$\begin{cases} -2x_1 - 2x_2 + 2x_3 - 2x_4 = -6 \\ -2x_1 - 2x_2 + x_3 = 1 \\ 6x_1 + 7x_2 - 8x_3 + 5x_4 = 11 \\ -6x_1 - 8x_2 + 9x_3 = 11 \end{cases}$$

1. **Write** the system in its matrix form (provide the matrix A and the vector B).
   **Solution:**
   Matrix $A$ **(0.5 pts)** and vector $B$ **(0.25 pts)** are:
   $$A = \begin{bmatrix} -2 & -2 & 2 & -2 \\ -2 & -2 & 1 & 0 \\ 6 & 7 & -8 & 5 \\ -6 & -8 & 9 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} -6 \\ 1 \\ 11 \\ 11 \end{bmatrix}$$

2. **Use** the Gaussian elimination method with partial pivoting to transform the system into an equivalent triangular system (swap with any row if the pivot is zero).
   **Solution:**
   We start by eliminating the sub diagonal elements of the first column, we take the following steps:

- **Step 1:** we multiply the first row by 1 and subtract it from the second row. **(0.25 pts)**
- **Step 2:** we multiply the first row by -3 and subtract it from the third row. **(0.25 pts)**
- **Step 3:** we multiply the first row by 3 and subtract it from the fourth row. **(0.25 pts)**

The resulting matrix is:

$$\begin{bmatrix} -2 & -2 & 2 & -2 \\ 0 & 0 & -1 & 2 \\ 0 & 1 & -2 & -1 \\ 0 & -2 & 3 & 6 \end{bmatrix}, \quad \begin{bmatrix} -6 \\ 7 \\ -7 \\ 29 \end{bmatrix}$$

We continue by eliminating the sub diagonal elements of the second column, in this case the pivot is zero, so we swap the second row with the third row **(0.75 pts)**:

$$\begin{bmatrix} -2 & -2 & 2 & -2 \\ 0 & 1 & -2 & -1 \\ 0 & 0 & -1 & 2 \\ 0 & -2 & 3 & 6 \end{bmatrix}, \quad \begin{bmatrix} -6 \\ -7 \\ 7 \\ 29 \end{bmatrix}$$

It is also possible to swap the fourth row with the second row, the gausian elimination result will not be the same, however the system solution will be the same. I consider any swap and linear combination of rows that leads to an equivalent triangular system as a valid answer.

Now we can proceed to eliminate the sub diagonal elements of the second column:

- We multiply the second row by -2 and subtract it from the fourth row **(0.5 pts)**.

The resulting matrix is:

$$\begin{bmatrix} -2 & -2 & 2 & -2 \\ 0 & 1 & -2 & -1 \\ 0 & 0 & -1 & 2 \\ 0 & 0 & -1 & 4 \end{bmatrix}, \quad \begin{bmatrix} -6 \\ -7 \\ 7 \\ 15 \end{bmatrix}$$

Finally, we can eliminate the sub diagonal element of the third column:

- We multiply the third row by 1 and subtract it from the fourth row **(0.5 pts)**.

The resulting matrix is:

$$\begin{bmatrix} -2 & -2 & 2 & -2 \\ 0 & 1 & -2 & -1 \\ 0 & 0 & -1 & 2 \\ 0 & 0 & 0 & 2 \end{bmatrix}, \quad \begin{bmatrix} -6 \\ -7 \\ 7 \\ 8 \end{bmatrix}$$

3. **Calculate** the solution of the obtained system.
   **Solution:**
   Using back substitution:

   - $x_4 = b_4/a_{44} = 8/2 = 4$ **(0.25 pts)**
   - $x_3 = (b_3 - a_{34}x_4)/a_{33} = (7 - 2 \times 4)/-1 = -1/-1 = 1$ **(0.25 pts)**
   - $x_2 = (b_2 - a_{23}x_3 - a_{24}x_4)/a_{22} = (-7 + 2 \times 1 + 1 \times 4)/1$ **(0.25 pts)** $= -7 + 2 + 4 = -1$ **(0.25 pts)**
   - $x_1 = (b_1 - a_{12}x_2 - a_{13}x_3 - a_{14}x_4)/a_{11}$ **(0.25 pts)**
     $= (-6 + 2 \times (-1) - 2 \times 1 + 2 \times 4)/-(2)$ **(0.25 pts)** $= (-6 - 2 - 2 + 8)/(-2) = (-2)/(-2) = 1$ **(0.25 pts)**

   The solution is

$$\begin{bmatrix} 1 \\ -1 \\ 1 \\ 4 \end{bmatrix}$$

4. **Verify** that the solution is valid for the original system.
   **Solution:**
   Substituting $x_1 = 1, x_2 = -1, x_3 = 1, x_4 = 4$ into the original system confirms that all equations are satisfied.
   $-2x_1 - 2x_2 + 2x_3 - 2x_4 = -6??$
   $-2(1) - 2(-1) + 2(1) - 2(4) = -2 + 2 + 2 - 8 = -6$ **(0.25 pts)**
   $-2x_1 - 2x_2 + x_3 = 1??$
   $-2(1) - 2(-1) + 1 = -2 + 2 + 1 = 1$ **(0.25 pts)**
   $6x_1 + 7x_2 - 8x_3 + 5x_4 = 11??$
   $6(1) + 7(-1) - 8(1) + 5(4) = 6 - 7 - 8 + 20 = 11$ **(0.25 pts)**
   $-6x_1 - 8x_2 + 9x_3 = 11??$
   $-6(1) - 8(-1) + 9(1) = -6 + 8 + 9 = 11$ **(0.25 pts)**
   All equations are satisfied, so the solution is valid.

## Exercise 3 - MI: (5 pts)

The following recursive function is supposed to calculate the determinant of a square matrix A using the cofactor expansion along the first row. However, the code contains some errors.

```
function d = determinant(A)
    [n, m] = length(A);
    if n ~= m
        error('The matrix must be square');
    end
    if n == 1 % Base case
        d = A(1, 1) * A(2, 2) - A(1, 2) * A(2, 1);
    else      % Recurrence relation
        for j = 1:n
            A[i, j] = [];
            d = d + (-1)^(1+j) * A(1, j) * determinant(A);
        end
    end
end
```

- **Find**, **explain**, and **correct** each error so that the function works correctly.
  **Solution:**
  Errors and corrections:

  1. **Error 1:** using `length` for the matrix dimensions **(0.25 pts)**, `size` should be used instead **(0.25 pts)**.
  2. **Error 2:** the base case is incorrect **(0.25 pts)**, the determinant of a $1 \times 1$ matrix is the element itself **(0.25 pts)**. Otherwise, the base case should be when the matrix is $2 \times 2$ (either correction is valid).
  3. **Error 3:** the variable `d` is not initialized **(0.25 pts)**, it should be initialized to 0 before the loop **(0.25 pts)**.
  4. **Error 4:** the submatrix is not correctly computed **(0.5 pts)**, it should exclude the first row and the $j^{th}$ column **(0.5 pts)**.

  **Corrected Code:**

```
function d = determinant(A)
    [n, m] = size(A);                                          % 0.5 pts
    if n ~= m
        error('The matrix must be square');
    end
    if n == 1
        d = A(1, 1);                                           % 0.5 pts
    else
        d = 0;                                                 % 0.5 pts
        for j = 1:n
            submatrix = A(2:end, [1:j-1, j+1:end]);            % 0.5 pts
            d = d + (-1)^(1+j) * A(1, j) * determinant(submatrix); % 0.5 pts
        end
    end
end
```