Assignment - 4
Lab Assignment

CSL3020: Computer Architecture
AY 2024-25, Semester – V
Due on: 15-09-2024

Total:100 Marks

General Instructions:

1. **Clearly mention the assumptions you have made, if any.**
2. **Clearly report any resources you have used while attempting the assignment.**
3. **Any submission received in another format or after the deadline will not be evaluated.**
4. **Make sure to add references to the resources that you have used while attempting the assignment.**
5. **Plagiarism of any kind will not be tolerated and will result in zero marks.**

Submission Guidelines:

1. Submit a single report depicting methods, results, and observations. Preparing a report is mandatory; failing it will lead to non-evaluation of the assignment.

2. Name your report as YourRollNo.pdf. And your program codes as yourRollNo.asm

3. There is no need to make a zip file. Just upload the report and program directly on Google Classroom, that is, the submission will contain{YourRollNo.pdf, YourRollNo.asm}. Do not upload files in any other format.

4. Do not copy-paste screenshots, etc. in the report. The report should look like a technical document, containing plots, tables, etc. whenever necessary.

5. Adhere to the instructions given, failing them may result in a penalty.

6. Late submissions will be penalized with 25% per day after the

deadline.

Objective:

The purpose of this lab assignment is to familiarize students with floating-point arithmetic and subroutines.

Instructions:

1. **Software Installation**:
    - **MARS (MIPS Assembler and Runtime Simulator)**:
        1. Visit the official MARS website.
        2. Download the latest version of MARS (`MARS4_5.jar`).
        3. Ensure you have Java installed on your system. If not, download and install Java from here.
        4. Run the `MARS4_5.jar` file by double-clicking it or using the command line: `java -jar MARS4_5.jar`.

Task:

Subtask 1:Catalan Number Calculation using Recursion: (50) The Catalan numbers form a sequence of natural numbers that have many applications in combinatorial mathematics, such as counting the number of valid combinations of parentheses, paths in grids, and binary search trees. The nth Catalan number can be calculated using the recursive formula:

$$C_n = \begin{cases} 1 & \text{if } n = 0 \\ \sum_{i-0}^{n-1} C_i \times C_{n-1-i} & \text{if } n > 0 \end{cases}$$

Tasks:

1. Main Program:

- Write an MIPS program to calculate the nth Catalan number using a recursive approach.
- The program should:
    - Take as input an integer n, where $n \geq 0$.
    - Use a subroutine to recursively compute C_n based on the provided formula.
    - Store intermediate Catalan numbers in memory to avoid redundant calculations (optional optimization).
    - Output the nth Catalan number.

2. Subroutine: Recursive Calculation of Catalan Numbers:

- This subroutine should:
    - Take the integer n as an input.

- If $n = 0$, return 1.0 as Co.
- If $n > 0$, calculate the sum $\sum_{i=0}^{n-1} C_{(i)} \times C_{(n-i-i)}$ recursively.
- Return the computed value $C_{(n)}$.

3. Input and Output:

- **Input:** The program should prompt the user to enter the value of n.
- **Output:** After computing the Catalan number, the program should display the nth Catalan number.

Subtask 2: Calculating the Exponential Function Using Taylor Series: (50) The exponential function e^x can be approximated using the Taylor series expansion:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!}$$

Tasks:

1. **Main Program:**
   - Write an MIPS program that calculates an approximation of e^x using the Taylor series expansion up to n terms.
   - The program should:
     - Take as input a floating-point number x and an integer n, which represents the number of terms in the series.
     - Use a subroutine to calculate each term in the series and add it to the running total.
2. **Subroutine: Compute the kth Term of the Series:**
   - This subroutine should:
     - Calculate the Kth term $\frac{x^k}{k!}$ of the series.
     - Use floating-point arithmetic to handle the calculation of both x^k and $k!$ (factorial).
     - Return the computed term as a floating-point value.
3. **Input and Output:**
   - The program should prompt the user to enter the value of x and the number of terms n.
   - After calculating the sum, the program should output the approximation of e^x as a floating-point value.

Deliverables:

- MIPS assembly code (submitted as a .asm file).
- A brief report detailed explanation of how the program works and its functionality.
- Your report should contain a detailed summary of each section of your code which should reflect your understanding.

The report should contain the following points:

- Provide a brief explanation of the MIPS code, focusing on its objective and key operations.
- Highlight any challenges faced and how you overcame them.
- Summarize the program's output.
- Reflect on what you learned from the assignment and the relevance of MIPS programming in real-world scenarios.

Evaluation Criteria

- Correctness: Accurate execution and analysis of the MIPS code.
- Comprehensiveness: Thorough analysis of output and methods
- Clarity: Clear and well-organized report presentation.
- Insightfulness: Depth of understanding and interpretation of performance data.

Note:

- **You are required to submit a report and code for both tasks before the deadline.**
- **Keep the code and screenshots of tasks from the lab with you for adding in the report.**