HOMEWORK 4: FUNCTION CALLS IN ARM
CPE221

Instructor: Rahul Bhadani

Due: March 21, 2025, 11:59 PM
100 points

You are allowed to use a generative model-based AI tool for your assignment. However, you must submit an accompanying reflection report detailing how you used the AI tool, the specific query you made, and how it improved your understanding of the subject. You are also required to submit screenshots of your conversation with any large language model (LLM) or equivalent conversational AI, clearly showing the prompts and your login avatar. Some conversational AIs provide a way to share a conversation link, and such a link is desirable for authenticity. Failure to do so may result in actions taken in compliance with the plagiarism policy.

Additionally, you must include your thoughts on how you would approach the assignment if such a tool were not available. Failure to provide a reflection report for every assignment where an AI tool is used may result in a penalty, and subsequent actions will be taken in line with the plagiarism policy.

Submission instruction:

Upload a .pdf on Canvas with the format {firstname_lastname}_CPE221_hw04.pdf. For example, if your name is Sam Wells, your file name should be sam_wells_CPE221_hw04.pdf. If there is a programming assignment, then you should include your source code along with your PDF files in a zip file {firstname_lastname}_CPE221_hw04.zip. Your submission must contain your name, and UAH Charger ID or the UAH email address. Please number your pages as well.

1 Assembly Program for Function Calls to Sort Numbers (30 Points)

Write an ARM assembly program that:

1. **(10 points)** Initializes an array x with the following integers:

$$\{1, 2, 4, 5, 3, 6, 8, 7, 10, 9\}.$$

2. **(20 points)** Divides the integers in x into two separate arrays, y and z:

   - The array y should store all even integers from x.
   - The array z should store all odd integers from x.

3. **(30 points)** Use a function named `intsort` to perform the sorting. The function must:

   - Take as input pointers to arrays x, y, and z, the index i, and the size of the array x.
   - Loop through the elements of x and separate even and odd integers using modulus operations.
   - Increment the appropriate counters (j for y, k for z) as each value is stored in y or z.

4. **(20 points)** At the end of execution:

   - y should contain all even numbers from x in the same order as they appear in x.
   - z should contain all odd numbers from x in the same order as they appear in x.

5. **(10 points)** Ensure that proper stack usage is maintained in the function `intsort` to save and restore the values of registers.

Additional Requirements (10 points)

- Use ARM assembly directives to define and reserve memory for the arrays and variables.
- Ensure the program uses proper branching instructions (B, CMP, etc.) to handle loop conditions and function calls.
- Include comments in your code to explain the logic behind each section.

2   Recursive Function in ARM (20 Points)

Write a recursive ARMv7 function to compute the factorial of a number. Use the stack to save and restore registers.

3   Nested Function Calls (20 Points)

Write a program in ARMv7 that computes `max(min(a,b,d,),f,g)` by nested function call. You must use frame pointers, stack pointers, stack frame.

4   C++ to ARM (20 Points)

Given the following C++ Code demonstrating a nested function call, write an equivalent ARMv7 code:

```
void inner() { x = 10; }
void outer() {
    int x = 5;
    inner();
}
```

You must also write a program that calls the `outer` function with appropriate value.

Write down the value of the link register (LR) at as the execution progresses.