HOMEWORK 5: STACKS AND FRAMES
CPE221

Instructor: Rahul Bhadani

Due: March 31, 2025, 11:59 PM
35 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_hw05.pdf. For example, if your name is Sam Wells, your file name should be sam.wells_CPE221_hw05.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_hw05.zip. Your submission must contain your name, and UAH Charger ID or UAH email address. Please number your pages as well.

---

1 Stacks and Frames

Consider the code given in C++. Write its corresponding implementation in the ARM assembly language. You should use stacks to store the local variable upon entering the function with the help of the frame pointers and stack pointers. Initialize the stack with 0x2000 address. **(25 Points)**.

Further, write down the stack status right before the function's epilogue. **(10 Points)**.

1.1 C Code:

```cpp
#include <iostream>
using namespace std;
int mpy_ne(int, int);
int abs(int);
int main()
{
    int first = -11;
    int second = -5;
    int result;
    result = mpy_ne(first, second);
}
int mpy_ne (int num1, int num2)
{
    int a, b, mult, i;
    a = abs(num1);
    b = abs(num2);
    mult = 0;
    for (i = 0; i < a; i++)
    mult = mult + b;
    if (num1 < 0) mult = - mult;
    if (num2 < 0) mult = - mult;
    return mult;
}
int abs(int x)
{
    if (x < 0) x = -x;
        return x;
}
```