

CS250 Lab 06 – Control Flow and Function Call in ARM (Take-Home)

DUE DATE FOR EVERYONE: March 21, 2016 at 11.59 pm

In this lab you will write 4 ARM assembly programs using the Raspberry Pi. You will learn how to write ARM subroutines (functions) and how to write ARM programs that can be called from a C program. C codes are provided where necessary. **This is a two weeks take-home lab. No late work will be accepted after the deadline, as there is Spring break after it.**

Below is the description for each ARM program:

Program 1: String length

Write a program with the code file name “mystrlen.s” that reads a string from stdin and prints the length of the string. You cannot use the C library function “strlen” in this assignment. The string is guaranteed to have no white space.

Sample output:

```
pi@raspberrypi:~/Desktop/cs250lab6/tests$ ./mystrlen
HelloTHERETHISIsCs250lab6
25
pi@raspberrypi:~/Desktop/cs250lab6/tests$
```

Program 2: Substring

You are given main_subString.c, which will call your ARM subroutine.

Your task is to finish subString.s, which locates the first occurrence of the substring in the source string and returns a pointer to the beginning of the first occurrence. If no such substring exists in source string, a null pointer is returned. The source string and the sub string are passed as arguments from the C program main_subString.c. Please check main_subString.c for more details.

The C program will first ask you to enter the source string, and then the substring, which you should locate in the source string.

Sample output:

```
pi@raspberrypi:~/Desktop/cs250lab6/tests$ ./subString
Enter the source string:long long ago in a galaxy far far away...
Enter the substring:galaxy
Returned String:galaxy far far away...
pi@raspberrypi:~/Desktop/cs250lab6/tests$
```

Program 3: Arithmetic computation

Write a program with the code file name “compute.s” that performs the basic arithmetic operations (addition, subtraction, multiplication and division). The program reads two integers from the stdin as well as the operator, and computes the result.

Sample output:

```
pi@raspberrypi:~/Desktop/lab6/tests$ ./compute
Enter the first number:5
Enter the operator:*
Enter the second number:6
Returned answer:30
pi@raspberrypi:~/Desktop/lab6/tests$
```

Program 4: Bubble sort

You are given main_bubble.c, which will call your ARM subroutine. Your task is to finish bubble.s, which bubble sorts an array of integers. This array and its length are passed as arguments from the C program main_bubble.c. Please check main_bubble.c for more details.

The C program will first ask you to enter an integer n, which is the number of integers you want to input, and then followed by n separate numbers, each one is an integer in your array.

Sample output:

```
pi@raspberrypi:~/Desktop/cs250lab6/tests$ ./bubble
Enter the length of your array:6
Enter the numbers in your array:
7
5
2
99
3
56
After bubble sorting:
2
3
5
7
56
99
pi@raspberrypi:~/Desktop/cs250lab6/tests$
```

Below is the bubble sort algorithm for reference:

```
procedure bubbleSort( A : list of sortable items )
  n = length(A)
  repeat
    swapped = false
    for i = 1 to n-1 inclusive do
      if A[i-1] > A[i] then
        swap(A[i-1], A[i])
        swapped = true
      end if
    end for
    n = n - 1
  until not swapped
end procedure
```

Submission: Turnin

Put your programs named as `mystrlen.s`, `subString.s`, `compute.s`, and `bubble.s` into a folder named `lab6-<yourPUIID>`. Turn in this folder using the ‘turnin’ commands as follows:

```
$ turnin -c cs250 -p lab6 lab6-<yourPUIID>
```

```
$ turnin -c cs250 -p lab6 -v
```

Grading

- You may use `gcc -S` as a learning tool, but you have to write your own programs by hand in assembly. You will receive no credit for assembly programs generated by `gcc`.
- A testall file is provided to check the correctness of your ARM programs.
- Credits will be given based on functionality of your program. Each program is tested with test cases. Check the grading rubric for points’ distribution.