Typical Questions for Experiment 3

Checkpoint 1

- What is the hex representation of `-1` as a 32–bit number?
- What is the decimal representation of `0xEF00234E`? What assumption did you make for conversion to decimal?
- What is the difference between `bgt` and `bhi` instruction?
- What are the ASCII values for characters 'Z' and 'z'?

Checkpoint 2

- What is the difference between `arr[6]` and `*(arr + 6)`?
- Explain why in the C program *matrix-c.c* neighbouring array elements are separated by 1, whereas in the assembly version of the program; *matrix-v1.s* the neighbouring array elements are separated by 4?
- What is the difference between the instruction "`ldr r6, [r2], #4`", and the instruction "`ldr r6, [r2, #4]`"?
- Why does the string "12345" seem to be stored in *reverse* as "4321" "5"?

Checkpoint 3

- Which registers are used to pass parameters to a function?
- Which register is used to return the result?
- What is the difference between the instruction "`bl`" and instruction "`b`"?

Checkpoint 4

- What is the major bug in the program *larger-v1.s* in Figure 12?
- What is the purpose of stack?
- What is the difference between a *full ascending* stack, an *empty ascending* stack, a *full descending* stack and an *empty descending* stack?
- What is the purpose of the *ARM Thumb Procedure Call Standard* was defined?
- What is the meaning of the instruction "`stmfd sp!, {fp, ip, lr, pc}`"?

Checkpoint 5

- How do you modify the function *iter-div.c* in Figure 15 to compute remainder as well as the quotient?

Checkpoint 6

- What are the values of a and b in function $f(x, y) = a + ((x/y) \times b)$?

Checkpoint 7

- Why is positional division more efficient than the iterative one?

Checkpoint 8

- What is the number of the executed assembly language instructions if all the `if`-statements are true?
- What is the number of the executed assembly language instructions if none of the `if`-statements are true?

Checkpoint 9

- What is the purpose of the object code *cstart.o*?

Checkpoint 10 (No Credit)

- How many lines code are there in the program of *posn-div-v1.c*?

Checkpoint 11 (No Credit)

- How much bigger your assembly code is compared with your program of *posn-div-v1.s*?

Checkpoint 12 (Extra Credit × 5)

- How much time did you spend in writing this program?