

Tests & Quizzes

Quiz 10

[Return to Assessment List](#)

Part 1 of 3 / 6.0 Points

 Question 1 of 3 6.0 Points

Using only 3 ARM instructions without using any LDR or STR instructions, show how to store the current value of the frame pointer in the stack, make the *frame pointer* to point to the base of this frame, **and** create an 8-byte stack frame **while** the *stack pointer* to point to the top of the stack. Assume that an **FD** stack is in use, appropriate stack space is already allocated to the stack, and the stack pointer is appropriately initialized.

 The first instruction is ✗ MOV SP, FP

 The second instruction is ✓ MOV FP, SP

 The third instruction is ✓ SUB SP, SP, #8

Answer Key: STMFD sp!,{fp} | STMFD r13!,{fp} | STMFD sp!,{r11} | STMFD r13!,{r11} | STMDB sp!,{fp} | STMDB r13!,{fp} | STMDB sp!,{r11} | STMDB r13!,{r11}, MOV fp,sp | MOV r11,sp | MOV fp,r13 | MOV r11,r13, SUB sp,sp,#8 | SUB r13,r13,#8 | SUB sp,#8 | SUB r13,#8 | SUB sp,sp,#0x8 | SUB r13,r13,#0x8 | SUB sp,#0x8 | SUB r13,#0x8

Part 2 of 3 / 4.0 Points

 Question 2 of 3 4.0 Points

Using only 2 ARM instructions without using any LDR or STR, show how to collapse the above-created 8-byte stack frame and restore the original values of the *frame pointer* and the *stack pointer*. Assume that an **FD** stack is in use.

The first instruction is ✗

 The second instruction is ✗ ADD R13, R13, #8

Answer Key: MOV sp,fp | MOV r13,fp | MOV sp,r11 | MOV r13,r11 | ADD sp,sp,#8 | ADD r13,r13,#8 | ADD sp,#8 | ADD r13,#8 | ADD sp,sp,#0x8 | ADD r13,r13,#0x8 | ADD sp,#0x8 | ADD r13,#0x8, LDMFD sp!,{fp} | LDMFD r13!,{fp} | LDMFD sp!,{r11} | LDMFD r13!,{r11} | LDMIA sp!,{fp} | LDMIA r13!,{fp} | LDMIA sp!,{r11} | LDMIA r13!,{r11}

Part 3 of 3 / 10.0 Points

Question 3 of 3 10.0 Points

Assume that you are writing an ARM assembly program, and this program will call `my_fun(int x, int *x)` function. Before calling the function, you need to push the parameters onto the stack.

You are only allowed to use `r0` as a working register to save any temporary value in it during the pushing operation if needed.

You are not allowed to use `LDM` or `STM` instructions in this question.

If you write any constant inside the instructions, you need to use decimal numbers only without any leading (insignificant) zeros.

The parameter `x` that you will use is located at address `FP + 8`.

The stack in this program is a **Full Ascending** stack, its space is *appropriately* allocated, and the `SP` is *appropriately initialized*.

Write 2 ARM assembly instructions to push `x` onto the stack.

The 1st instruction is: ✓ LDR R0, [FP, #8]

The 2nd instruction is: ✗ STR R0, [SP]

Write 2 ARM assembly instructions to push `*x` onto the stack.

The 1st instruction is: ✗ MOV R0, [FP, #8]

The 2nd instruction is: ✗ STR R0, [SP]

Answer Key: LDR r0,[fp,#8]|LDR r0,[r11,#8], STR r0,[sp,#4]!|STR r0,[r13,#4]!, ADD r0,fp,#8 | SUB r0,fp,#-8|ADD r0,r11,#8 | SUB r0,r11,#-8, STR r0,[sp,#4]!|STR r0,[r13,#4]!