Question **1**

Not yet answered

Marked out of 1.00

WHICH OPERATION IS USED TO TOGGLE THE VALUE OF BITS?

Select one:

- a. BIC
- b. AND
- c. ORR
- d. None of the answers
- e. EOR

Clear my choice

Question **2**

Not yet answered

Marked out of 1.00

GIVEN AN INTEGER VARIABLE A. HOW TO CLEAR BIT 2 OF VARIABLE A IN C? THE REMAINING BITS SHOULD BE UNMODIFIED.

Select one:

- a. None of the answers
- b. a &=~(1<<2);
- c. a|=1<<2;
- d. a|=~(1<<2);
- e. a&=1<<2;

Clear my choice

Question **3**

Not yet answered

Marked out of 1.00

WHICH OF THE FOLLOWING IS WORD-ALIGNED ADDRESS?

Select one:

- a. 0x80000001
- b. 0x80000003
- c. 0x80000008

Question **4**

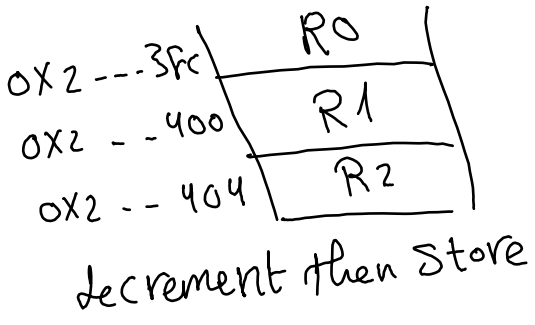
Not yet answered

Marked out of 1.00

ASSUMING SP = 0x20000408, WHAT IS THE VALUE OF SP AFTER THE INSTRUCTION PUSH {R0-R2}?

Select one:

- a. 0x20000400
- b. 0x20000203
- c. 0x200003FC
- d. None of answers
- e. 0x20000404



Question **5**

Not yet answered

Marked out of 1.00

IN ARM CORTEX M4, WHAT IS THE ROLE OF R15?

Select one:

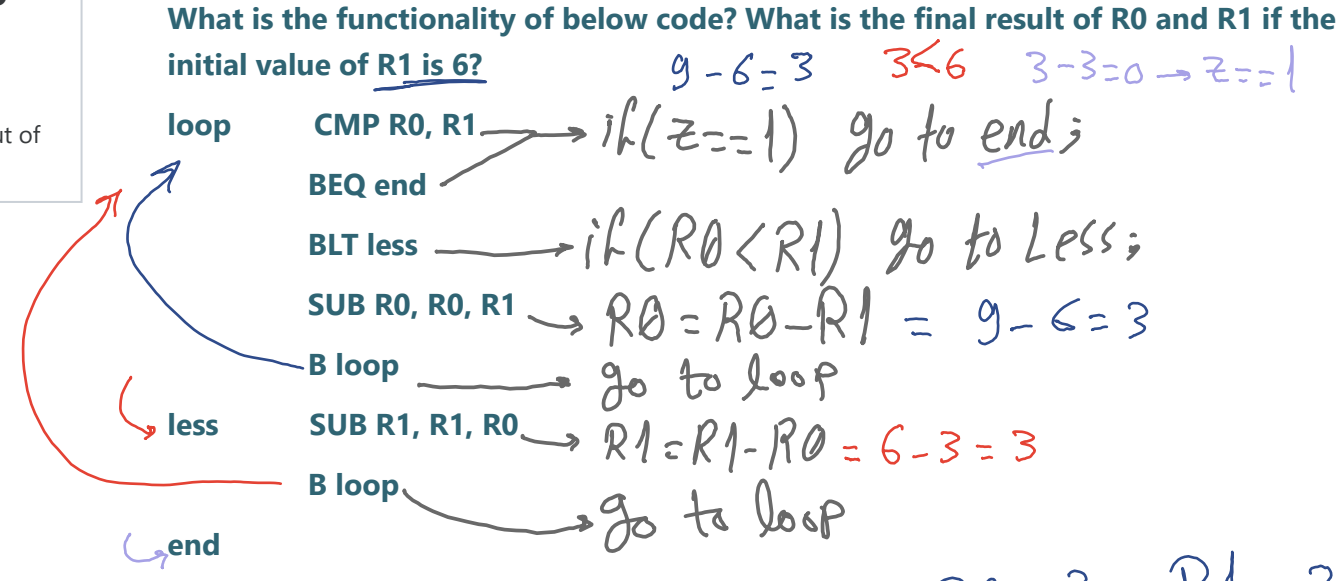
- a. Condition Bits Register
- b. None of the answers
- c. Link Register
- d. Program Counter
- e. Stack Pointer

Question **6**

Not yet answered

Marked out of 3.00

What is the functionality of below code? What is the final result of R0 and R1 if the initial value of R0 is 9 and the initial value of R1 is 6?



Select one:

- a. None of the answers
- b. Greatest Common Divisor, R0=3 , R1= 3
- c. Highest Common Factor, R0=6, R1=6
- d. Least common Multiple, R0=9, R1=18
- e. Calculating logarithm of a number to base 2, R0=3, R1=3

Question **7**

Not yet answered

Marked out of 1.00

WHAT IS THE PURPOSE OF THE C FLAG IN THE PSR OF CORTEX-M PROCESSORS?

Select one:

- a. The C flag is set in case of an unsigned overflow
- b. The C flag is set if the result of the operation is less than zero
- c. The C flag is set if the result of the operation is zero
- d. None of the other answers

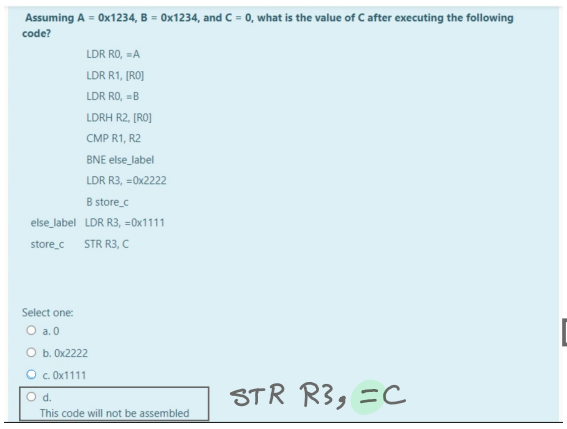
Question **8**

Not yet answered

Marked out of 1.00

Assuming A = 0x1234, B = 0x1234, and C = 0, what is the value of C after executing the following code?

```
LDR R0, =A
LDR R1, [R0]
LDR R0, =B
LDRH R2, [R0]
LDR R5,=C
CMP R1, R2
BNE else_label
LDR R3, =0x2222
B store_c
else_label  LDR R3, =0x1111
store_c     STR R3, [R5]
```



Select one:

- a. 0x2222
- b. 0x1111
- c. This code will not be assembled
- d. 0
- e. None of the answers

Question **9**

Not yet answered

Marked out of 1.00

ASSUMING WE WANT TO LOAD 2 BYTES FROM THE ADDRESS STORED IN R1 INTO R0, WHICH INSTRUCTION TO USE?

Select one:

- a. LDR R0, [R1]
- b. LDR R0, =R1
- c. None of the answers
- d. LDRH R0, [R1]
- e. LDRB R0, [R1]

Question **10**

Not yet answered

Marked out of 1.00

HOW MANY REGISTERS DO THE ARM CORTEX-M PROCESSORS HAVE?

Select one:

- a. None of the other answers
- b. 16
- c. 15
- d. 13

Question **11**

Not yet answered

Marked out of 1.00

WHICH INSTRUCTION WILL BE USED?
ASSUME WE WANT TO MOVE AN IMMEDIATE 32-BIT NUMBER (0X22222222) INTO R1.

Select one:

- a. LDR R1, =0x22222222
- b. MOV R1, =0x22222222
- c. None of the other answers
- d. LDR R1, #0x22222222

Question **12**

Not yet answered

Marked out of 1.00

Big Endian Byte Order is the **most significant** byte (the "big end") of the data is placed at the byte with the lowest address. The rest of the data is placed in order in the next three bytes in memory.

Example: A variable X with value 0x01234567 will be stored in address 0x100 as
@addr-->value
0x100 --> 0x01
0x101 --> 0x23
0x102 --> 0x45
0x103--> 0x67

Based on what explained above, trace following instructions, assume list start at memory location 0x0000018 and using ARM Big Endian. What is the result of **R0 and R2** after execution?

LDR R0, =LIST

MOV R10, #0x2

LDR R2, [R0, #4]! R0 = R0 + 4 R2 = Mem[R0]

AREA READ_variables, DATA, READONLY

LIST DCB 0x34, 0xF5, 0x32, 0xE5, 0x01, 0x02,0x8, 0xFE

Select one:

- a. R0= 0x18 and R2= 0x010208FE
- b. None of the answers
- c. R0= 0x1C and R2= 0x34F532E5
- d. R0= 0x18 and R2= 0xFE080201
- e. R0= 0x18 and R2= 0x34F532E5
- f. R0= 0x1C and R2= 0xFE080201
- g. R0= 0x1C and R2= 0x010208FE

↳ 0x18 + 0x4

Question **13**

Not yet answered

Marked out of 1.00

Big Endian Byte Order is the **most significant** byte (the "big end") of the data is placed at the byte with the lowest address. The rest of the data is placed in order in the next three bytes in memory.

Example: A variable X with value 0x01234567 will be stored in address 0x100 as
@addr-->value
0x100 --> 0x01
0x101 --> 0x23
0x102 --> 0x45
0x103--> 0x67

Based on what explained above, trace following instructions, assume list start at memory location 0x0000018 and using ARM Big Endian. What is the result of **R1** after execution?

LDR R0, =LIST R0 = &LIST

MOV R10, #0x2

LDR R1, [R0] → R1 = Mem[R0]

AREA READ_variables, DATA, READONLY

LIST DCB 0x34, 0xF5, 0x32, 0xE5, 0x01, 0x02,0x8, 0xFE

Select one:

- a. R1= 0xE532F534
- b. R1=0 x34
- c. None of the above
- d. R1=0x34F532E5
- e. R1=0x010208FE

Question **14**

Not yet answered

Marked out of 1.00

WHAT IS CONTENTS OF R1 IN THE BELOW INSTRUCTION?
ASSUME R1= 0x80008001.
LSR R1,R1,#3

Select one:

- a. 0xC0004000
- b. 0x00080010
- c. None of the answers
- d. 0x10001000

Question **15**

Not yet answered

Marked out of 1.00

* **WHAT IS THE ADDRESSING MODE OF THE INSTRUCTION LDR R0, [R1]?**

↳ Regular Register indirect

Select one:

- a. PC-relative Addressing
- b. Indexed Addressing
- c. None of the answers
- d. Immediate Addressing
- e. Direct Addressing

Question **16**

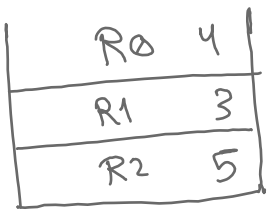
Not yet answered

Marked out of 1.00

ASSUMING THE TOP 3 VALUES OF THE STACK IN ORDER ARE 4, 3, AND 5 WHAT IS THE VALUE OF REGISTER R0 AFTER THE INSTRUCTION POP {R0-R2}?

Select one:

- a. None of the answers
- b. 4
- c. 5
- d. 3



Question **17**

Not yet answered

Marked out of 1.00

WHICH OPERATION IS USED TO CLEAR THE VALUE OF BITS?

Select one:

- a. AND
- b. None of the answers
- c. OR
- d. EOR
- e. BIC

*Why AND not BIC? the main operation to clear bits is And

The BIC instruction performs an AND operation on the bits in *Rn* with the complements of the corresponding bits in the value of *Operand2*.

Question **18**

Not yet answered

Marked out of 1.00

From the below program, what are the values of R4, R5, and R7 after the execution of the program?

```
        MOV  r4, #7      R4 = 7
        MOV  r5, #3      R5 = 3
        MOV  r6, #3      R6 = 3
Again   MOV  r7, r4      R7 = R4 = 7 → = 10 → = 17
        ADD  r4, r5, r4  R4 = R5 + R4 = 3 + 7 = 10 → = 7 + 10 = 17 → 10 + 17 = 27
        MOV  r5, r7      R5 = R7 = 7 → = 10 → = 17
        SUBS r6, r6, #1  R6 = R6 - 1 = 3 - 1 = 2 → = 2 - 1 = 1 → 1 - 1 = 0
                                                                  ↑ Z == 1
        BNE Again → if (Z == 0) go to Again;
```

R4 = 27 = 0X1B R5 = 17 = 0X11 R7 = 17 = 0X11

Select one:

- a. R4=0x1B, R5=0x11, R7=0x11
- b. R4=0x03, R5=0x03, R7=0x03
- c. None of the answers
- d. R4=0x04, R5=0x04, R7=0x03
- e. R4=0x04, R5=0x03, R7=0x02

Question **19**

Not yet answered

Marked out of 1.00

WHAT DOES HAPPEN WHEN RESET IS EXECUTED IN ARM-CORTEX M BASED MICROCONTROLLER?

Select one:

- a. The 32-bit value stored at location 0 of flash ROM is loaded into the SP and the 32-bit value stored at location 4 of flash ROM is loaded into PC and LR register value is set to 0xFFFFFFF0.
- b. The 32-bit value stored at location 0 of flash ROM is loaded into the PC and the 32-bit value stored at location 4 of flash ROM is loaded into SP and LR register value is set to 0xFFFFFFFF.
- c. The 32-bit value stored at location 0 of flash ROM is loaded into the SP and the 32-bit value stored at location 4 of flash ROM is loaded into PC and LR register value is set to 0xFFFFFFFF.
- d. None of the answers

Question **20**

Not yet answered

Marked out of 1.00

GIVEN AN INTEGER VARIABLE A. HOW TO SET BIT 2 OF VARIABLE A IN C? THE REMAINING BITS SHOULD BE UNMODIFIED.

Select one:

- a. a|=1>> 2;
- b. a|=1>>3;
- c. a|=1<< 2;
- d. a|=1<< 3;
- e. None of the answers

Question **21**

Not yet answered

Marked out of 1.00

WHAT IS THE VALUE OF R0 AND R1 AT THE END OF THE PROGRAM?

```
AREA WRITE_variables, DATA, READWRITE
z  DCD 0
        AREA MYCODE, CODE, READONLY
LDR r4, =a
LDR r0, [r4]
LSL r0, r0, #2
LDR r4, =b
LDR r1, [r4]
AND r1, r1, #15
ORR r1, r0, r1
LDR r4, =z
STR r1, [r4]
B   END_LOC
a       DCD 1
b       DCD 18
END_LOC NOP
```

R4 = a
R0 = 4 * R0 = 4 * 1 = 4
R1 = b
... 001111 10010 01111
R1 = R1 & 15 = 18 & 15 = 00010
01111 & 10010 = 00010
100 OR 010 = 110
R1 = R0 | R1 = 4 | 2 = ob 110 = 6
R4 = &z
Mem[R4] = R1
directives implemented before implementing the program.
like #define but its stored in memory.

Select one:

- a. 3, 6
- b. 2, 5
- c. 3, 5
- d. None of the answers
- e. 4, 6

Question **22**

Not yet answered

Marked out of 1.00

Big Endian Byte Order is the **most significant** byte (the "big end") of the data is placed at the byte with the lowest address. The rest of the data is placed in order in the next three bytes in memory.

Example: A variable X with value 0x01234567 will be stored in address 0x100 as
@addr-->value
0x100 --> 0x01
0x101 --> 0x23
0x102 --> 0x45
0x103--> 0x67

Based on what explained above, trace following instructions, assume list start at memory location 0x0000018 and using ARM Big Endian. What is the result of **R0 and R5** after execution?

LDR R0, =LIST R0 = &LIST

MOV R10, #0x2

LDRB R5, [R0], #1 R5 = (1byte Mem[R0]) then R0 = R0 + 1

AREA READ_variables, DATA, READONLY

LIST DCB 0x34, 0xF5, 0x32, 0xE5, 0x01, 0x02,0x8, 0xFE

R5 = 0x34 R0 = 0x19

Select one:

- a. R0= 0x19, R5=0x34
- b. R0= 0x19, R5=0xE532F534
- c. R0= 0x19, R5=0x34F532E5
- d. R0= 0x18, R5=0x34F532E5
- e. None of the answers
- f. R0= 0x18, R5=0x34

◄ Assignment 2 Jump to... CSE Quiz 1 (sections 3 and 4) ►

*More Practice:

FROM THE BELOW PROGRAM, WHAT ARE THE VALUES OF R4, R5, AND R7 AFTER THE EXECUTION OF THE PROGRAM?

MOV R4, #7 → R4 = 7
MOV R5, #5 → R5 = 5
MOV R6, #5 → R6 = 5
AGAIN MOV R7, R4 → R7 = R4 = 7 → = 12 → = 19 → = 31 → = 50
ADD R4, R5, R4 → R4 = R5 + R4 = 5 + 7 = 12 = 7 + 12 = 19 → 12 + 19 = 31 = 19 + 31 = 50 = 5 0 + 31 = 81
MOV R5, R7 → R5 = R7 = 7 = 12 → = 19 → 31 → 50
SUBS R6, R6, #1 → R6 = R6 - 1 = 5 - 1 = 4 → 3 → 2 → 1 → 0
BNE AGAIN → if (z==0) goto Again

A) R4=0x04, R5=0x03, R7=0x03

B) R4=0x51, R5=0x32, R7=0x32

C) R4=0x04, R5=0x04, R7=0x03

D) R4=0x03, R5=0x03, R7=0x03

E) None of the answers

ANSWER: B

R4 = 81 = 0x51 , R5 = 50 = 0x32 , R7 = 50 = 0x32