**Module 2: The Arm Cortex-M0 Processor Architecture: Part 1**

1. Which Arm processor family should be your first choice if you are developing a wearable healthcare monitoring device?

1. Cortex-A series
2. Cortex-R series
3. Cortex-M series
4. Cortex-SC series

2. Which of the following properties is not a feature of the Cortex-M0 architecture?

1. Energy efficiency
2. Low silicon costs
3. Faster software development and reuse
4. Enhanced security

3. Which Arm processor family should be your first choice if you are developing an automotive braking system?

1. Cortex-A series
2. Cortex-R series
3. Cortex-M series
4. Cortex-SC series

4. Which Arm processor family should be your first choice if you are designing a digital TV?

1. Cortex-A series
2. Cortex-R series
3. Cortex-M series
4. Cortex-SC series

5. Which Cortex-M processor is not based on Von Neumann architecture?

1. Cortex-M0
2. Cortex-M0+
3. Cortex-M1
4. Cortex-M7

6. Which statement explains a difference between the Arm architecture and an Arm processor?

1. Unlike the Arm architecture, Arm processors do not usually include an exception model.
2. Unlike Arm processors, the Arm architecture usually has detailed timing information.
3. Unlike the Arm architecture, Arm processors usually include detailed hardware structure.
4. The Arm architecture and an Arm processor are identical with no differences.

7. What is the maximum number of instructions that can be executed concurrently by a Cortex-M0 processor?

1. One
2. Fifty-six
3. Three
4. Nine

8. Which of the following tasks is performed by the wakeup interrupt controller (WIC) block?

1. It performs the function of interrupt masking while the NVIC and the processor core are in sleep mode.
2. It compares the priority between interrupt requests and the current priority level.
3. It communicates with the processor so that the processor can execute the correct interrupt handler.
4. It handles interrupt request signals coming from the non-maskable interrupt (NMI) input.

9. Which of the following is not stored in R15? (There may be more than one correct answer.)

1. A memory address for a constant
2. The address of the next assembly instruction that will be executed
3. The memory’s address that holds the last value stored on the stack
4. The application program status register (APSR) negative flag

10. Which of the following memory regions can you use to execute instructions stored in their address space?

1. External RAM region
2. External device region
3. Peripheral region
4. None of the above.