Problem 1 Multiple Choice
50 points

You get 2 points for each correct answer and 0 points for no answer. Wrong answer results in -0.66 points. Only one alternative is correct.

a) Which of the following does not apply to embedded systems/microcontrollers?

1. They usually have a tight power budget.
2. They are often used in applications with real time requirements.
3. They are most useful and commonly used to run applications with very large memory footprints.
4. They may work in extreme environments such as satellites.

b) What is *false* regarding immediate values in assembly code, and ARM Thumb II in particular?

1. Their size may vary depending on the instruction type.
2. Shorter instructions (e.g., 16 vs 32 bits) typically mean shorter fields to store immediate values.
3. All instructions in an ISA necessarily contain one or more immediate value.
4. Multiple instructions may be required to load a register from immediate values.

c) Regarding the conditional instruction *ite*:

1. It enables the conditional execution of other instructions without using branches.
2. It is an optimized branch instruction that only inserts one delay slot instead of two.
3. It cannot be implemented in a pipelined processor.
4. It is especially useful for long branches and should be avoided for short branches.

d) When calling the following function using the Procedure Call Standard for the ARM Architecture or the ARM Procedure Call Standard, what of the following statements is *false*:

1. Registers r0-r3 are used to pass parameters into the procedure.
2. Register r0 is used to hold a return value.
3. Register r11 stores the link register
4. Register r13 stores the stack pointer

e) What of the following is a common way to pass parameters when calling a subrutine:

1. By storing them in the registers and the heap.
2. By storing them in the registers and pushing them in the stack.
3. By storing their values in the .data segment.
4. By storing their values in the .bss segment.

f) A stack frame or activation register stores:

1. Local variables.
2. Global variables.
3. Dynamically allocated variables.
4. Executable code.

g) The application binary interface or ABI is a standard that:

1. Aims at making architectures with different ISAs able to execute the same binaries.
2. Makes binaries work across different operating systems by just using different dynamic libraries.
3. Enables programming language portability, e.g., C code can run in a Java-like VM such as Dalvik.
4. Determines how subrouines, data structs, etc. are handled so that different binaries can work together.

h) A disadvantage of using a frame pointer is:

1. The stack cannot be examined by a debugger.
2. It requires support from the caches, and it is expensive.
3. It makes a register unavailable for general use.
4. It makes stack frames larger.

i) GPIO pins provide a flexible interface between a microncontroller and the outside world. However, one inconvenience of using GPIO for this is:

1. They can only work as output pins.
2. They can only work as input pins to implement the so called bit banging.
3. They typically require using the CPU to run protocols implemented in SW.
4. It is not possible to provide interrupts when a pin changes state, requiring the CPU to poll the pin state directly.

j) Regarding I/O device access:

1. Only disk devices can use memory mapping.
2. The most common access method is encoding data and commands through IRQ lines.
3. When using a separate I/O address space it is impossible to use IRQs.
4. I/O instructions and memory mapping can be both present in the same processor.

k) Which of the following statements about processor modes is true:

1. Both user mode and supervisor mode enable access to all resources in the machine.
2. The same ADD instruction is typically slower in supervisor mode than in user mode because privileged accesses require runtime checks for safe execution.
3. Supervisor modes are typically used to run the operating system kernel to prevent it from eavesdropping on user programs.
4. User modes have restricted access so that we can provide protection between user programs.

l) Banked registers:

1. Provide faster switches between processor modes.
2. Enable faster execution of arithmetic instructions.
3. Simplify the internal circuitry of the CPU.
4. Require that values are pushed in the stack to save all multiple copies of the registers when entering another processor mode.

m) Which of the following is *false* regarding set-associative and fully-associative caches, assuming they can store the same number of blocks:

1. Fully-associative caches always require more (or the same) number of tag comparisons per access.
2. A direct mapped cache has more different indexes than a set-associative cache.
3. A direct mapped cache needs a replacement policy such as LRU or random.
4. Fully associative caches do not suffer conflict misses.

n) In a modern system with virtual memory, which of the following is a *false* statement:

1. The Memory Management Unit (MMU) translates virtual addresses issued by the processor to physical addresses accessible in memory.
2. Memory is divided into fixed size blocks, called pages.
3. Translating a memory address in a modern system always requires one or several accesses to main memory to read the page table.
4. The page offset is the same in the virtual and in the physical addresses.

o) Assume the following assembly code containing 5 instructions (i1 to i5) is executed in a 3-stage pipeline (fetch, decode, execute) able to start one instruction per cycle. What of the statements below is *false* about the code and its execution in this pipeline:

i1 - sub r1, r2, r3

i2 - add r2, r1, r3

i3 - sub r1, r2, r3

i4 - add r7, r9, r8

i5 - sub r9, r2, r3

1. There is a RAW hazard between i2 and i5.
2. There is a RAW hazard between i1 and i2.
3. There is a data dependence between i2 and i5.
4. There is a name dependence between i4 and i5.

p) One of the reasons why the I^2C bus is widely used in industry is:

1. Slaves never introduce wait states by design, making it a fast and simple alternative to more complex protocols.
2. It uses a simple protocol and only requires two wires.
3. It uses a fast round-robin arbitration mechanism in which collisions are not possible.
4. It is faster than other buses such as AXI.

q) The main differences between static RAM (SRAM) and dynamic RAM (DRAM) are:

1. SRAM is made of transistors only while DRAM is made of capacitors only.
2. DRAM requires more transistors per bit than SRAM, enabling higher storage densities.
3. DRAM requires periodic refreshes because the transistors lose their charge due to leakage currents.
4. SRAM requires more transistors per bit than DRAM.

r) The main reason to use procedure inlining is:

1. To reduce the size of the final code.
2. To save the overhead of procedure calls.
3. To increase compilation speed.
4. To increase the cache hit rate.

s) Given the following C code:

```
N = 128;
for(i = 0; i < N; i++)
   for(j = 0; j < N; j++) {
      A[i][j] = A[i][j] + i + j;
      B[j][i] = B[j][i] + i * j;
   };
```

Consider array accesses only (A, B) and ignore any other accesses. The elements of A and B have a size of 4 bytes. Given a 16KB direct mapped cache with 64 byte blocks, initially empty, which of the following statements is true:

1. Only cold cache misses will take place.
2. Loop tiling using blocks of 16x16 elements plus an array padding of 1KB between A and B does not avoid all conflict and capacity misses.
3. All conflict and capacity misses can be avoided by just inserting the right padding space between A and B.
4. Loop fission can be used to avoid all conflict misses.

t) When choosing between an oscilloscope and a logic analyzer, what of the following applies:

1. An oscilloscope is more suited to analyze large sequences of logic values due to its analog nature.
2. A logic analyzer can provide a more accurate analog signal shape.
3. A logic analyzer displays 0 and 1 values, making the memory footprint of each stored sample lower.
4. A logic analyzer is more suited to detect signal quality faults.

u) Regarding the power/performance trade-offs of adjusting the voltage and frequency of a CPU with DVFS, which of the following statements is true:

1. Increasing frequency increases power consumption, always resulting in more energy spent to execute the same program faster.
2. Decreasing voltage decreases dynamic power consumption quadratically, but it also reduces the maximum frequency achievable, which makes more energy always necessary to execute the same program more slowly.
3. Dynamic power consumption depends on both frequency and voltage, but static power consumption depends on frequency only.
4. Static power consumption can make frequency reductions increase the overall energy consumed to execute a program.

v) Energy harvesting can be used to power microncontrollers, with the following consideration:

1. The only possible way to prevent a brown-out is to harvest enough energy from the environment to keep the microncontroller always on.
2. Vibrations have prevented piezoelectric materials from being used to power microncontrollers, as they would break them.
3. An energy bank may be required to store the harvested energy, and even when such energy bank is used, we must have mechanisms to avoid brown-outs.
4. Brown-outs can never happen when supplying energy to the microncontroller by means of solar cells.

w) Microncontrollers may contain one or several caches to improve performance, but regarding energy efficiency:

1. Caches always increase the energy consumed because of the static power they add.
2. Cache accesses are more expensive than accessing DRAM memory, so high cache hit rates should be avoided.
3. Turning caches off when the observed cache hit rate is very low can save energy.
4. Going to a deep sleep mode that power gates the caches keeps the cache contents and saves energy.

x) In the context of process scheduling, what of the following situations is possible in a preemptive priority based scheduling and has the elements necessary to be described as priority inversion:

1. A process gets exclusive access to a shared resource, resulting in the starvation of two other processes.
2. A high priority process cannot run in the CPU because a low priority process is running in the CPU.
3. A high priority process waits for a low priority process to release a shared resource while a medium priority process runs in the CPU.
4. A medium priority process gets hold of the CPU because a high priority process yields it, starving a low priority process.

y) Assume we are using rate monolitic scheduling. Three processes, A,B and C contend for the CPU. Their periods (and deadlines) are 3s, 4s and 6s, respectively. They run for exactly 1, 2 and 1 seconds each time they execute, respectively:

1. All deadlines are met because the utilization of the CPU of the 3 processes is 100% (1/3 + 2/4 + 1/6 = 1).
2. Process A fails to meet its second deadline (t = 6s).
3. Only process C will ever miss deadlines, assuming no context switch overhead.
4. Earliest deadline first scheduling would meet all deadlines, even assuming realistic context switch overheads.