Solver: Chen Yinya
Email Address: ychen063@e.ntu.edu.sg

1. (a) Speedup = time old/ time new = 1 / (0.6/12 + 0.4/4) = 6.667

   (b) Dynamic power consumption $P=ACV^2f$
   A, C remain unchanged, V becomes half
   Maximum possible frequency is proportional to V, thus f is halved
   Percentage change = $(ACV^2f$ new - ACV^2f old)/ ACV^2f old = $(0.5^3-1)/1$= -87.5% which means dynamic power consumption decrease 87.5%

   Static power consumption $P=VI_{leak}$
   Percentage change = $(0.5 - 1)/1 = -50\%$

   (c) (i) To toggle the bits, we should xor the bits with 1. INSTR is xor. To flip the bits at position 12, 8, 4, 0, the bits of s1 at position 12, 8, 4, 0 should be 1. Thus, #0xZYXV = #0x1111.

   (ii) Insert 2 NOPs between I2 and I3. – RAW for S2
   Insert 2 NOPs between I3 and I4. – RAW for S0
   Insert 2 NOPs between I4 and I5. – RAW for S0
   Insert 2 NOPs between I6 and I7. – RAW for S2
   Insert 2 NOPs between I6 and I7. – RAW for S2
   with forwarding, 2 NOPs should be inserted after I7 for control hazard

   (iii) With full data forwarding, 1 stall cycle needs to be inserted between I3 and I4. PC is updated at the execute stage, there is 1 stall cycle inserted after BNE branch instruction.
   S2 decrement by 4 (FFFC) every loop. Initially s2 = 0x00C0 = 192. Number of loops is 192 /4 = 48.
   Total number of stall cycles = (1+1) *48 = 96
   In every loop, 5 instructions and 2 stalls
   Steady state CPI = (No of instructions + No of stalls) / No of instructions = (5+2)/5=1.4

2. (a) Unconditional jump to an absolute address should use J offset instruction. nPC is FFFAC = 1111 1111 1111 1010 1100
   Using J Offset, offset should be new PC divided by 4. Move the address to the right by 2 bits. Offset = 11 1111 1111 1110 1011 = 3FFEB (hex) = 262123 (dec)
   The instruction should be J 262123
   There range of Jump instruction can cover is from 0 to 111…111(26 bits). The min address is using offset 0, final address is 0x50000000.
   The max address is using offset 3FFFFFF, final address is 0x5FFFFFFC.

(b) assume list length is given in $s2.

```
        lw $v0, 0($s1) # initialize maximum to A[0]
        addi $s0, $zero, 0 # s0 stores index
        addi $a0, $zero, 0
loop: addi $ s0, $ s0, 1
        beq $s0, $s2, done
        addi $a0, $v1, $zero
        lw $a1, 4($s1)
        jal max-2
        addi $s1, $s1, 4
        j loop
done la $s3, 0x00100304
 sw $v0, 0($s3)
```

(c) 1. Data hazard. The source or destination register of an instruction are not available at the time expected in the pipeline. For example,
I1: SUB $t3, $t2, $t1;
I2: AND $t5, $t3, $t4;
t3 will cause data hazard in this case.

2. Control hazard. Conditional and unconditional jumps, subroutine calls, and other program control instructions can stall a pipeline because of a delay in the availability of an instruction.
For example,

```
        BEQ $s1,$s2, loop
Loop: ADD $t1, $t2, $t3
```

After the fetch stage of BEQ instruction, it is not decided yet to branch or not.
This is control hazard.

3. Structural hazard is caused by two instructions require to use of a given hardware resource at the same time. For example, one instruction is writing to memory another one is reading from memory.

3. (a)

For N-way set-associative cache, # set = Total cache size / (Block size x N)

For 8-way set-associative cache, # set = 128KB / (128 x 8) = 2^17/(2^7 x 2^3) = 2^7.

8-way – set index = $\log_2$ 2^7 = 7 bits
Block offset = $\log_2$ 128 = 7 bits
Tag = 32 (address space) - 7 - 7 = 18 bits

(b) (i) AMAT = Time for a hit + Miss rate x Miss penalty
64 bytes miss penalty = 80 + 64/16 *2 = 88 cycles
To calculate X: 1.93 = 1 + X% * 88, X = 1.06
To calculate Y: Y = 1+ 2.64% * 88, Y= 3.32
To calculate Z: Z = 1+ 0.51% * 88, Z = 1.45

(ii) 64 bytes has smaller AMAT for cache size of 256
For cache size from 4KB to 64KB, block size 64 bytes has smaller miss rate. For cache size 256KB, block size 256 bytes has smaller miss rate.

(iii) Average memory access time (AMAT) is more important. It is a common metric to analyze memory system performance. When miss rate of A is considerably higher than B, average memory access time of A can still be smaller than B with shorter clock cycles and less overhead.

4. (a) Two policies are write back and write through. Write back is used more often.

(b)(i) Virtual page 0,3,5,7 are not in memory. The virtual addresses are 0-1023, 3072-4095, 5120-6143, 7168-8191
(ii) Virtual address 1024 is the first byte of virtual page 1. Page 1 has frame number 0. The physical address is 0. Virtual address 3071 is the last byte of virtual page 2. Page 2 has frame number 2. The physical address is 3071.

(c)(i) ASIC has the best performance per unit power consumption. FPGA has the best programmability.
(ii) ARM big.LITTLE is a heterogeneous computing architecture developed by ARM Holdings, coupling relatively battery-saving and slower processor cores (LITTLE) with relatively more powerful and power-hungry ones (big).

All the best for your exams!