Question 1 (30 points):

1. (**10 points**) Let `i` be an integer value stored in memory at the address given by `$sp+8`. `P` is a vector of 32-bit integers, and `C` is a vector of characters. Assume that the address of `P` is in `$a0` and the address of `C` is in `$a1`. Write the segment of MIPS assembly instructions to execute the following C-language statement:
   `C[P[i]+3] = C[P[i+1]];`
   Use the minimum number of instructions (a solution that uses nine instructions is known).

```
lw  $t0, 8($sp)         # $t0 <- i
sll $t1, $t0, 2         # $t1 <- 4*i
add $t2, $a0, $t1       # Address(P[i])
lw  $t3, 0($t1)         # $t3 <- P[i]
lw  $t4, 4($t1)         # $t4 <- P[i+1]
add $t5, $a1, $t4       # $t5 <- Address(C[P[i+1]])
lb  $t6, 0($t5)         # $t6 < C[P[i+1]]
add $t7, $a1, $t3       # Address of (C[P[i]])
sb  $t4, 3($t2)         # C[P[i]+3] <- C[P[i+1]]
```

2. (**5 points**) Assume that `p` is in the memory position given by `$fp-4` and `q` is in the memory position whose address is `$fp-8`. Assume that in this machine addresses have 32 bits, and that `short` is a 16-bit integer. Write MIPS assembly code for the statement in the following segment of C-language code:

```
short *p;
short *q;
*p = *(q+1);
```

   Use the minimum number of instructions (a solution that uses four instructions is known).

```
lw  $t0, -8($fp)        # $t0 <- q
lh  $t1, 2($t0)         # $t1 <- *(q+1)
lw  $t2, -4($fp)        # $t1 <- p
sh  $t1, ($t2)          # *p <- $t1
```

3. (**5 points**) Assume that the jump instruction at address 0xCFFF FFFC has the following binary representation: `0x0800 0CC0`. Which is the address of the target of this instruction?

```
PC+4   = 0xD000 0000
target = 0xD000 3300
```

4. (**10 points**) The Intel Itanium processor is an IA64 architecture processor. When running in this architecture the Microsoft Windows operating system uses a page size of 8KB. The L1 instruction cache can store 16 KB of data, is 4-way set associative and each cache block contains 32 bytes. The Itanium operates with a 64-bit address bus. Assume that a given machine constructed with the Itanium implements 64GB of physical RAM memory. Given an instruction fetch to a specific address in this machine, specify how many bits are used for each of the following elements of the memory hierarchy:

   64GB = 2^{36}, therefore the physical address has 36 bits

   (a) Virtual Page Number: 64-13 = 51 bits

   (b) Physical Page Number: 36-13 = 23 bits

   (c) Virtual Page Offset: 13 bits

   (d) Physical Page Offset: 13 bits

   (e) L1 Icache offset: 5 bits

   (f) L1 Icache index: 7 bits

   Each set in the cache has 4×32 bytes = 128 bytes.
   Therefore there are $\frac{16KB}{128} = \frac{2^{14}}{2^7} = 2^7$ sets.

   (g) L1 Icache tag: 64-7-5 = 52 bits