1. (6 points total) Consider the following declarations compiled for an x86-64 machine running Linux:

```c
struct node {
    char c[5];
    int i;
    short j;
    struct node *next;
} my_node;
```

(a) _____ How many bytes are allocated for each instance of `my_node`?

A. 8  B. 18  C. 20  D. 24  E. 32  F. none of these

(b) _____ How many bytes are in the smallest struct containing the same elements (but reordered) as `my_node`?

A. 8  B. 18  C. 20  D. 24  E. 32  F. none of these

(c) _____ Suppose `my_node` were a union instead of a struct. How many bytes would be allocated for each instance?

A. 8  B. 18  C. 20  D. 24  E. 32  F. none of these

2. (2 points) Suppose we invent a new technology that allows storing 3 states, not just 2. Instead of bits (0, 1), we have trits (-1, 0, +1). What’s the maximum number of integers we could represent in n trits?

A. $2^n$  B. $3n$  C. $3!$ (3 factorial)  D. $3^n$  E. $2^{n+3}$
3. ______ (2 points) Arguments passed to functions in x86-64 are passed via:
   A. main memory
   B. the stack
   C. registers
   D. a combination of stack and registers
   E. None of the above.

4. ______ (2 points) With respect to byte ordering, the x86 is
   A. little endian
   B. big endian
   C. has no “endianess”
   D. has “endianess” determined by the operating system

5. ______ (2 points) Assume that %rcx has value a and %rsi has value b. Under which of the following conditions is the branch to .L5 taken?
   - subq %rsi,%rcx
   - jg .L5
   A. a < b       B. a ≤ b       C. a > b       D. a ≥ b

6. ______ (2 points) On an x86-64 Linux system, which of these take up the most bytes in memory?
   A. char a[7]
   B. float d
   C. short b[3]
   D. int *c

7. ______ (2 points) What is the C equivalent of movq 0x10(%rax, %rcx, 4), %rdx?
   A. rdx = *(rax + rcx*4 + 0x10)
   B. rdx = *(rax + rcx + 4 + 0x10)
   C. rdx = rax + rcx + 4 + 10
   D. *(rax + rcx + 4 + 10) = rdx

Page total: ______/10