

CS 255 - Computer Organization/Architecture I
Practice Questions for Final

Note: I do not have answers to these questions. Nor do I intend to provide answers to them. You should work them out among yourselves. You should have learned all that is necessary from your project assignments already but if you still have problems with these questions, you may ask me *specific questions*. I will not answer questions like "how do you do this one?" because I want to educate students and do not want to teach students how to prepare for tests...

Note: The Final is Comprehensive and covers *all* material discussed in the course. So you should not forget the practice material for the Midterm in the preparation for the final.

The final will ofcourse focus on material not covered by the midterm. These material include: recursion in ARM and all material on C.

Question 1.

Write the following recursive QuickSort function in ARM:

```
void QuickSort(int X[ ], int n1, int n2)
{
    int pivot, curr, right_array_spot;

    if (n1 < n2)
    {
        pivot = X[n2];
        right_array_spot = n2;
        curr = n1;

        while (curr != right_array_spot)
        { if (X[curr] <= pivot)
          { curr++;
            }
          else
          {
              X[right_array_spot] = X[curr];
              X[curr] = X[right_array_spot-1];
              right_array_spot--;
            }
          }

        X[curr] = pivot;
        QuickSort(X, n1, curr-1);
        QuickSort(X, curr+1, n2);
    }
}
```

Question 2.

Write the following program in ARM assembler:

```
class List
{
    Link next;
    int v1;
    int v2;
};

List head, ptr;
int sum;

sum = 0;
for (ptr = head; ptr != null; ptr = ptr.next)
    sum = sum + ptr.v1 + ptr.v2;
```

Question 3.

The following is the famous Euler algorithm to determine the Greatest Common Divisor (GCD) of 2 numbers x and y as a *recursive* algorithm:

```
int GCD( int x, int y )
{
    if ( y == x )
        return x;
    else
    {
        while ( x > y )
            x = x - y;
        return GCD( y, x );
    }
}
```

Translate the program into ARM assembler.

Question 4

The following 2 C program files (p1.c and p2.c) are compiled with the command

```
gcc p1.c p2.c
```

C program file 1 (p1.c): =====	C program file 2 (p2.c): =====
<pre>extern int a; void f1() { a = 1; // (1) int a; // (2) a = 1; } void f2(int a) { { int a; // (3) a = 1; } } void f3(int a) { int a; // (4) a = 1; } int a; static int b;</pre>	<pre>void f4() { extern int a; a = 1; // (5) } void f5() { extern int b; b = 1; // (6) } int main(int argc, char *argv[]) { a = 1; // (7) f1(); f2(1); f3(1); f4(); f5(); }</pre>

Questions:

- Will the C compiler report an error for line (1) ? (Yes / No)
- Will the C compiler report an error for line (2) ? (Yes / No)
- Will the C compiler report an error for line (3) ? (Yes / No)
- Will the C compiler report an error for line (4) ? (Yes / No)
- Will the C compiler report an error for line (5) ? (Yes / No)
- Will the C compiler report an error for line (6) ? (Yes / No)
- Will the C compiler report an error for line (7) ? (Yes / No)

Question 5

A list structure is defined as follows:

```
struct List
{
    int value;
    struct List *next;
};
```

Question:

- Write the *recursive* function:

```
struct List * insertList( struct List *head, struct List *elem)
```

in C that inserts a List element `elem` in the list that starts at `head` and maintains an **ordered list**.

The function must return the head element of the *new* list.