ECE198KL – Exam 2 practice questions

C control and iteration constructs

1. Implement the function `void replace_spaces(char *str, int length)` in C which replaces all spaces in the string 'str' with the characters '%20'. Note that you cannot create a new string. You have to modify the contents of 'str'. This type of algorithms are called In-place algorithms. 'length' represents the length of the input array.

Eg:
Before calling your function: str = “ECE198KL is awesome!”
After calling your function: str = “ECE198KL%20is%20awesome!”

We have allocated enough memory to hold the resultant string.

2. Implement the function `void reverse_word_order(char *str, int length)` in C which reverses the order of words in the string 'str'. A word is defined as a set of characters separated by a space.

Eg:
Before calling your function: str = “ECE198KL is super cool!”
After calling your function: str = “cool! super is ECE198KL”

Note that only the order of words is reversed but not the word itself.

Hint: One way to solve is to reverse the entire string and then the letters of each individual word. Can you solve it using a stack? +5 if you can :D

Functions – stacks, frames, activation record

1. Convert the C function below to an LC-3 assembly language subroutine. While converting from C to LC-3, use the runtime stack and activation record implementation described in the textbook (R5 is the frame pointer, R6 is the stack pointer, etc). You may assume that the input values for parameters alpha and beta are positive integers (greater than zero). You may also assume that the passed parameters are already saved in the activation record on the runtime stack (this was done by the calling program).

```
int multiplyAdd(int a, int b)
{
int factor = 0;
int ans = 0;
factor = 3;
ans = alpha * beta + gamma;
return ans;
}
```

Arrays and pointers

1. Remember in Program 6, where you had to access an element at [i, j] in a 2D array through a pointer as arr_ptr[i*WIDTH +j] where 'arr_ptr' is the pointer to the memory location that represents the 2D array and WIDTH is the no. of columns of the 2D array.
Using the same notation implement the function zero_matrix(int *matrix, int width) in C such that if any element in the array is 0, its entire row and column are set to 0.
'matrix' is always square meaning it has equal number of rows and columns. 'width' represents the number of rows (which is also columns) of the matrix.

You can create any many variables you like.

Hint: You may have to iterate the array twice.

Eg:
Before calling your function:
matrix (represented as)
0 1 2
3 4 5
6 7 8
After calling your function:
0 0 0
0 4 5
0 7 8
2. Given two sorted arrays, pointed to by 'a' and 'b', implement the function void merge_arrays(int *a, int *b, int len_a, int len_b, int *sorted), provided to you, in C such that it merges both the arrays so that the resultant array is sorted. 'len_a' and 'len_b' are lengths of input arrays. The sorted array should be stored in the memory location pointed to by 'sorted'. Enough memory is already allocated by us. You need not allocate any memory.

Eg:

Input arrays:
a → 2 5 6 9
b → 1 3 7 10

Output array:
sorted → 1 2 3 5 6 7 9 10

Short answers

1. The following function `double divide_by_2 (int num)` is trying to divide a number by 2. Do you think it is correctly implemented? If yes, explain. If not, correct the error and explain.

```
double divide(int num)
{
    int divisor = 2;
```

```
    return (num/divisor);
}
```

2. What is the difference between global and static variables?

3. What is the following function doing?

```
void function(char *s, char *t)
{
    while (*s++ = *t++)
         ;
}
```

4. Imagine you are working in a company and asked to write test cases for the following function which is comparing two strings. How would do go about it?

```
/* strcmp: return <0 if s < t, 0 if s = t, >0 if s > t */
int strcmp (char *s, char *t)
{
    for (; *s == *t; s++, t++) {
        if (*s == '\0')
            return 0;
    }
     return *s - *t;
}
```

What would you comment on the specification of the function?