CS100 Homework 2 (Fall Semester 2018)

Due time: 11:59 pm, October 28, 2018

Homework 2 comprises 3 problems. Please finish all of them independently.

1. Write a function **digitValue()** that returns the value of the *k*th digit (*k* = 1, 2, …) from the right of a non-negative integer *n*. If *k* is larger than the total number of digits in *n*, the function returns 0. For example, digitValue(123456, 3) returns 4, and digitValue(3158, 6) returns 0. The prototype of the function is given below:

   ```
   int digitValue(int n, int k);
   ```

   Write a C program to test the function.

   A sample input and output session is given below (the red and italic lines are user's input, in which the first line is the value of *n* and the second line is the value of *k*; the blue line is the program's output):

   123456
   3
   4

   Note: You should not use arrays or character strings.
   Please save your program code in a source file named "hw2_1.c".

2. Write a function that extracts the odd digits from a positive number *n*, and combines the odd digits sequentially into a new number. The new number is to be passed back to the calling function via **call by reference**. If the input number n does not contain any odd digit, then the output should be −1. For example, if *n* = 1234567, then 1357 should be the result; but if *n* = 28, then −1 is the result. Write the function, named **`extOddDigits()`**, that passes the result through the second parameter, *nd*. The function prototype is given as follows:

   ```
   void extOddDigits(long n, long *nd);
   ```

   Write a C program to the test your function.

   Three sample input and output sessions are given below (the red and italic lines are user's input, whereas the blue lines are the program's output):

   1234567
   1357

1357924
13579

246
−1

Note: You should not use arrays or character strings.
Please save your program code in a source file named "hw2_2.c".

3. Write the code for the following functions for operations on matrix:

```
void transpose(int M[SIZE][SIZE]);
      /* the function transposes a square matrix. */

void swap2Rows int M[SIZE][SIZE], int r1, int r2);
      /* the function swaps row M[r1] with row M[r2] */

void swap2Cols int M[SIZE][SIZE], int c1, int c2);
      /* the function swaps column M[][c1] with column M[][c2] */
```

Write a C program to test the above functions. In addition, your program should print the resultant matrix after each operation. You may assume that the input matrix is a 3× 3 matrix when testing the functions. Moreover, r1 = c1 = 1, and r2 = c2 = 2.

A sample input and output session is given below. The red and italic numbers are user's input (with two consecutive numbers in the same row separated by a blank space). The blue matrices are program's output, where the first blue matrix is the transpose of the input matrix, the second blue matrix is after swapping two rows of indices r1 and r2, and the third matrix is after swapping two columns of indices c1 and c2. The results are cumulative. In the printed output, the width of each column should be fixed to 6 characters.

```
5 10 15
15 20 25
25 30 35

5     15    25
10    20    30
15    25    35

5     15    25
15    25    35
10    20    30
```

```
5     25    15
15    35    25
10    30    20
```

Please save your program code in a source file named “hw2_3.c”.