

SOFTWARE—Ph.D. Qualifying Exam Fall 2009

(i) (7 pts.)

Consider a function called `is_identity_matrix` that has the following specification.

Given an $N \times N$ (square) matrix return true (1) if the matrix is an identity matrix, and false (0), otherwise. The “matrix” argument is a pointer to an array of integers that stores the elements of the matrix. The “size” argument gives the value of N .

Assume that $N > 1$. Assume also that the matrix is stored row-by-row in an N^2 -element array of integers. For example, the matrix

```
1 3 4
2 2 5
0 2 3
```

would be stored as the following sequence of integers (in successive elements of the array pointed to by “matrix”).

```
1 3 4 2 2 5 0 2 3
```

Note that an identity matrix is a square matrix in which all diagonal elements are 1 and all non-diagonal elements are 0.

A prototype for the function is given below.

```
int is_identity_matrix(int *matrix, int size);
```

Develop a complete C code implementation of the function `is_identity_matrix`.

Write your solution to Question (i) on this page. Please clearly indicate your solution and show all work on this page.

(ii) (7 pts.)

Consider the following structure definition for constructing linked lists of numbers.

```
/* Container for one element of a linked list of numbers. */
struct numbers_list_element {
    double number;
    struct numbers_list_element *next;
};
```

Consider also the following function prototype and associated header comment.

```
/******
Given a pointer to the first element ("head") of a linked list of
numbers, return the median value in the list of numbers. If the list is
empty ("head" is NULL), then print a descriptive error message to
standard error and return 0. Assume that upon entry to this function, the
list of numbers is sorted in increasing order (i.e., lower-valued numbers
appear earlier in the list than higher-valued numbers).
```

Recall that given an ordered sequence of numbers, the **median** is the number in the middle if the list has an odd number of elements, and the median is the average of the two numbers in the middle if the list has an even number of elements.

```
*****/
double median(struct numbers_list_element *head);
```

Develop a complete C code implementation of the function `median`.

Write your solution to Question (ii) on this page. Please clearly indicate your solution and show all work on this page.

(iii) (6 points) Write a function that takes two strings as arguments, and returns a string that is the result of interleaving characters from both of the argument strings. The argument strings must have the same length. If s_1, s_2, \dots, s_N and t_1, t_2, \dots, t_N are the sequences of characters in the first and second argument strings, respectively, then the function returns the interleaved string (“result string”) $s_1, t_1, s_2, t_2, \dots, s_N, t_N$. If $N = 0$ (both strings are empty), then the result string is an empty string. If the given strings (“s” and “t”) do not have the same length, then the results string is also empty. The return value of the function is a pointer to a dynamically allocated array that contains the result string.

Here is a prototype of the function:

```
char *interleave(char *s, char *t);
```

Develop a complete C code implementation of the function `interleave`.

Write your solution to Question (iii) on this page. Please clearly indicate your solution and show all work on this page.