\Diamond CSCI 2500 — Computer Organization \Diamond Fall 2019 Quiz 1 (September 11, 2019)

Name:	
RCS ID:	
Section:	
RIN#:	
Zone/Row#/Seat#:	

Please silence and put away all laptops, notes, books, phones, electronic devices, etc. This quiz is designed to take 50 minutes; therefore, for 50% extra time, the expected time is 1 hour and 15 minutes and 100% extra time is 1 hour and 40 minutes. Questions will not be answered except when there is a glaring mistake or ambiguity in the statement of a question. Please do your best to interpret and answer each question.

Name:	
I Controc.	

1. (10 POINTS) Suppose you are given the following code fragment:

```
void compute(int endpoint) {
  int nums[6][6];
  .
  .
  .
}
```

Where is the memory allocated for nums? Clearly circle the **best** answer.

(a) The runtime stack

(d) The runtime text segment

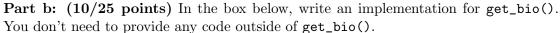
(b) The runtime heap

- (e) The runtime environment
- (c) The runtime code segment
- (f) Memory is not allocated for nums
- 2. (25 POINTS) You are writing function get_bio() that asks the user to enter their first name, graduation year, and GPA and makes all these values available to the caller function without using the return value. You may assume that memory allocation for character arrays is performed by the caller function.

Part a: (6/25 points) You start get_bio() definition by writing the header. Which of the following options would be the best choice? Clearly circle the best answer.

```
(a) void get_bio( char &f_name, int &grad_year, float &GPA )
(b) void get_bio( char *f_name, int grad_year, float *GPA )
(c) void get_bio( char &f_name, int *grad_year, float *GPA )
(d) void get_bio( char *f_name, int &grad_year, float &GPA )
(e) void get_bio( char *f_name, unsigned int grad_year, float GPA )
(f) void get_bio( char *f_name, int *grad_year, float *GPA )
(g) void get_bio( char *f_name, unsigned int &grad_year, float &GPA )
(h) void get_bio( char f_name, int grad_year, float GPA )
```

Name:	



Part c: (9/25 points) Finally, given the variables defined in the code fragment below, fill in the arguments for a call to get_bio():

```
int main(void) {
  char *name = (char *) malloc(100 * sizeof(char));
  int year;
  float gpa;
  /* Fill in the arguments below: */
  get_bio(
    .
    .
    return EXIT_SUCCESS;
}
```

Name:	

3. (20 POINTS) Given unsigned int variables n and m, write a fragment of C code to dynamically allocate the amount of memory needed to create a two-dimensional array containing $n \times m$ double values. Call this array arr and be sure to properly declare this variable. You don't need to initialize the double values in the array.				

Name:	
vame:	

use the following command. C (a) echo \$\$	learly circle the \mathbf{best} answer. (d) echo $\mathtt{SEXIT_CODE}$
(b) echo \$-	(e) echo \$!
(c) echo \$?	(f) echo \$snapchat
Part b: (10/15 points) You shell command you selected in	a run snapchat and try to analyze its behavior by using to part (a).
On the first run, the shell com	mand prints 0. What does it tell you about the program ru
	command prints 1. What does it tell you about the progra
On the second run, the shell orun?	command prints 1. What does it tell you about the progra
	command prints 1. What does it tell you about the progra
	command prints 1. What does it tell you about the progra
	command prints 1. What does it tell you about the progra
	command prints 1. What does it tell you about the progra

Name:	

5. (30 POINTS) The vowels() function below is supposed to take input string s and return a new string containing only the vowels in s. All other characters should be ignored. As an example, if s is "SnApChAt RuleS!!!" then the function returns "AAue" (i.e., a string of four bytes). Unfortunately, there are four errors in the code below. Find and correct each error. You may assume that all header files and the main() functions are properly defined and just not shown here. Do not simply rewrite the function or change the coding style used.

```
char * vowels( char * s )
  int i, j, k, count;
  char * result;
  char * v = "aeiouAEIOU";
  for (i = 0 ; i < strlen(s) ; i++)
     for (j = 0; j < strlen(v); j++)
       \mathbf{if} \ (\mathbf{s}[\mathbf{i}] = \mathbf{v}[\mathbf{i}])
          count++;
  result = malloc( count );
  for (i = 0, k = 0; i < strlen(s); i++)
     for (j = 0 ; j < strlen(v) ; j++)
       \mathbf{if} \ (\mathbf{s}[\mathbf{i}] = \mathbf{v}[\mathbf{j}])
          result[k++] = s[i];
  result[k] = " \setminus 0";
  return result;
}
```