CIS 1201 Finals Mock Exam (Retake) | TIME LIMIT: 1 HOUR 30 MINS

Place your answers at the designated areas of the answer sheet provided.

Test 1: Multiple choice.

1. Given the declaration: char string[] = "CIS 1201". Determine sizeof(string).
   A. 7 B. 8 C. 9 D. 1 E. None of the choices

2. What is the ASCII integer equivalent for a space character?
   A. 0 B. 10 C. 12 D. 32 E. None of the choices

3. Which of the following is syntactically incorrect given the declaration below?
   #define dollar '$'
   char a, b, x = 'A'
   A. a = x; B. if (x < 'x') b = x; C. a = dollar; D. a = 5 + '5'; E. b = "b";

4. The array component A[i] is equivalent to which of the following?
   A. a + i B. a++ C. *a + i D. (*a + i) E. None of the choices

5. Which character function checks whether the given character is a hexadecimal digit or not?
   A. ishex() B. ishdigit() C. isxdigit() D. None of the choices

6. Which character function checks whether the given character is a graphical character but not alphanumeric?
   A. ispunct() B. isgraph() C. issymbol() D. None of the choices

7. Which of the following operations can be done using structures but not in arrays?
   A. Whole data structure assignment using =
   B. Pass data structure to a function by copy and by address
   C. Return a local data structure to the calling function
   D. All of the above
   E. None of the above

8. What value will be returned by fclose() signifying a successful close operation?
   A. 0 B. 1 C. 2 D. Nothing to return E. None of the choices

9. A linked list with head pointer L is pointing to the first of 5 nodes, each node containing an integer and a pointer to the next node of the list. How many static variables are there?
   A. 1 B. 6 C. 11 D. 5 E. None of the choices

10. What value does myArr[2][1][0] in the sample code below contain?
    int myArr[3][2][2] = {1,2,3,4,5,6,7,8,9,10,11,12};
    A. 11 B. 9 C. 7 D. 5 E. None of the choices

11. If a structure S is pointer to by pointer P, which of the following is used to access a structure member M through P?
    A. P->S.M B. (*P).S.M C. (*P)->M D. (*P).M

Test 2. Fill in the blanks.

Evaluate the following:

11-12. 26 + 13 >> 2 << 2
13-14. 11 + 11 & -11 ^ 11
15-16. Find the minimum value of A such that 17 | A = 117.

Given the declaration:

```
int arr[5][5] = {{-1}, {-2, -3}, {-4, -5, -6}, {-7, -8, -9, -10}};
```

17-18. Determine sizeof(arr) + sizeof(arr[1]) + sizeof(arr[2][3]).
19. How many nonnegative elements are present in the array?

True or False:
20. arr == *arr
21. arr[2] == arr[2][0]

Assume that the value of *arr* is ABC000. Determine the following:
22. Value of arr[1][1]
23. Address of arr[1][1]
24. Value of arr + 2
25. Value of arr[2] + 2

Answer the following:
26-28. Declare a pointer variable *ptr* with an appropriate datatype that can be used to navigate through the 2D array above, then have it initialized to the beginning address of the second row of array *arr*.

29-30. After the statement above and then the statements

```
ptr++;
(*(*(ptr+1)+1))--;
```

Is a component of the array arr modified? (YES/NO)
If the answer is YES, write the value of the modified component; otherwise, write NO.

Test 3. Code fragment completion

Function specification: Function returnTranspose() will return the transposition of a given matrix given a row size, column size, and the contents of the matrix. The transpose of a matrix is an operator which flips a matrix over its diagonal; that is, it switches the row and column indices of the matrix A by producing another matrix. See example below.

31-40. Fill in the blanks below to complete the function definition.

Given:	To return:
1 2 3 4 5 6 7 8 9 0 1 2	1 5 9 2 6 0 3 7 1 4 8 2

```
int** returnTranspose(int row, int col, int arr[][col]){
   int a, b;
   int ** trans = ___(31-32)___;
   for (a=0; ___(33-34)___; a++){
      _______(35-36)_________;
      for (b=0; ___(37-38)___; b++){
         _______(39-40)_______;
      }
   }
   return trans;
}
```

Function specification: Function sortByCol() will sort the given 2D array in a way the values in a given column x will be in increasing order using bubble sort technique. See example below.

41-50. Fill in the blanks below to complete the function definition.

```
int stud[11][5];

1  10 20 30 40          2  20 18 39 12
2  20 18 39 12          1  10 20 30 40
3  28 31 38 11          5  11 23 19 72
4  26 32 91 28          7  53 27 19 11
5  11 23 19 72          9  25 28 28 26
6  76 35 10 27          3  28 31 38 11
7  53 27 19 11          4  26 32 91 28
8  34 82 29 19          6  76 35 10 27
9  25 28 28 26          11 13 39 71 39
10 73 62 82 83          10 73 62 82 83
11 13 39 71 39          8  34 82 29 19

sortbyCol(11,5,2,stud);
```

```
void sortByCol(int row, int col, int x, int arr[][col]){
   int a, b;
   ___(41-42)___; /*Declare temporary array to hold the
                    set of values during the swapping*/
   for(a=0; a<row-1; a++){
      for(b=0; b<row-a-1; b++){
         if(___(43-44)___){  /*Perform the swapping*/
            ___(45-46)___;
            ___(47-48)___;
            ___(49-50)___;
         }
      }
   }
}
```

Test 4. Programming

A mall consists of cinemas, each cinema shows a specific movie with its corresponding movie details and the current seating arrangement. Each seat stores the occupancy status and the details of the person occupying a seat if necessary. See the structure definitions below.

```
#define ROW 15
#define COL 30
#define SIZE 5

typedef struct{
   char fName[33];
   char MI;
   char lName[33];
}NameDetails;

typedef struct{
   NameDetails custName;
   enum{
      VACANT, OCCUPIED
   }status;
}SeatDetails;
```

```
typedef struct{
   int id;
   char movieName[33];
   float price;
}MovieDetails;

typedef struct{
   int row;
   int col;
}SeatLocation;

typedef struct{
   SeatDetails seats[ROW][COL];
   MovieDetails movie;
}CinemaSeating;
```

```
typedef struct{
   CinemaSeating Cinema[SIZE];
}Mall;


/*Linked list implementaion of
ticket purchase requests (TPR)*/

typedef struct TPR{
   NameDetails custName;
   SeatLocation loc;
   int movieID;
   struct TPR* next;
}*TPRList;
```

Create the following functions:

51-70. buyTickets() – Given a series of ticket purchase requests (TPR) and a pointer to a mall, this function will process every TPR by looking into the details of the request. For every TPR, the function shall reserve a seat for the given customer if the movie exists, if the seat exists and if the seat is vacant. If the reservation is successful, the node containing the corresponding TPR shall be removed from the list.

71-80. getTotalSales() – Given a pointer to a mall, the function will return to the calling function the total sales for the mall based on the occupied seats in every cinema.

-- END OF TEST | REVIEW YOUR ANSWERS! --