2020 Summer COP 3502 Section 2 Final Exam - Part A Solutions

1) (10 pts) A class has ***n*** students, and the i^{th} student has taken s_i tests. This information is stored in a file where the first line has the value of ***n*** and the following ***n*** lines have information about each student. On the i^{th} of these lines, the first value is s_i. This is followed by s_i integers, all in between 0 and 100, inclusive, representing the test scores. Here is a sample file:

```
4
10 100 80 90 100 90 90 95 80 100 95
3 87 93 90
5 100 90 90 100 90
8 90 90 90 90 80 80 85 85
```

Write a segment of code that reads in this information from standard input (assuming file redirection on the command prompt as shown in class) into an array of arrays that is dynamically allocated to have precisely the correct number of slots to store the data. Name your array studentscores. All necessary variables have been declared for you below. Please use only these in your solution. **Note: you are ONLY allocating the memory and reading in the information. No need to do anything else.** Also, the way this is written, you wouldn't have stored the length of each separate array, but don't worry about that for the purposes of this question.

Solution

```
int** studentscores;
int i, j, numStudents, numScores;

scanf("%d", &numStudents);
studentscores = malloc(numStudents*sizeof(int*));
for (int i=0; i<numStudents; i++) {
    scanf("%d", &numScores);
    studentscores[i] = malloc(numScores*sizeof(int));
    for (int j=0; j<numScores; j++)
        scanf("%d", &studentscores[i][j]);
}
```

Grading

1 pt - scanf numStudents, 2 pts - malloc studentscores, 1 pt for, 1 pt scanf, 2 pts malloc, 1 pt loop, 2 pts scanf

2) (10 pts) Write a function that takes in a pointer to a linked list of nodes storing integers and a variable named value, and returns the number of nodes in the list storing that value. For example, if a list pointed to by listPtr stores 2, 6, 2, 3, 4, 2, 6, and 6 and value = 6, your function should return 3, since 6 appears in the list 3 times. Please use the struct and function prototype provided below:

```
typedef struct node {
    int data;
    struct node* next;
} node;

int countInList(node* listPtr, int value) {
    if (listPtr == NULL) return 0;
    int res = 0;
    if (listPtr->data == value) res = 1;
    return res + countInList(listPtr->next, value);
}
```

Grading (recursive version): 2 pts base case, 4 pts rec call, 3 pts add in 0 or 1 appropriately, 1 pt return

```
int countInList(node* listPtr, int value) {
    int res = 0;
    while (listPtr != NULL) {
        if (listPtr->data == value)
            res++;
        listPtr = listPtr->next;
    }
    return res;
}
```

Grading (iterative version): 1 pt init var, 2 pts while, 3 pts if, 1 pt update res, 2 pts move ptr, 1 pt return

3) (5 pts) What is the value of the following post-fix expression? (Note: You will be graded solely on your final response.)

3 9 + 4 * 12 2 / / 2 8 5 - * +

The corresponding infix expression is

(((3 + 9) * 4) / (12/2)) + (2*(8 - 5))

The value of this expression is 48/6 + 2*3 = 8 + 6 = **14**

Grading: 5 pts for 14, 0 pts for all other answers. No need to show stack.