Answer the questions in the spaces provided on the page. If you run out of room for an answer, continue on the back of the page.

- **DO NOT OPEN THE EXAM UNTIL TOLD TO DO SO**
- You only need to answer the first question and 4 of the 5 remaining questions.
- On one of the questions, make a large slash across the page, which indicates that it should not be graded.
- On every page (including the first and last page), write your first and last name, before answering the question. Unnamed pages may be lost.
- If you start to answer a question and then change your mind, please cross out the attempt and write *DO NOT GRADE* across it.
- Legibility matters! If we can’t read your answer, you will receive a 0 for it.

![Flowchart: How to Write Good Code](https://xkcd.com/844/)
I need a program that can read in some input and store the information into some variables. My input is formatted as follows:

Name: Josh  
Class: 220  
Grade: 2.0

I know that no one in the class has a name that is more than 100 characters. Be sure that you don't allow names longer than 100 characters crash your program.

Here is my code:

```c
#include <stdio.h>
#define NAME_MAX_SIZE 100
int main(void) {
    int class;
    float grade;
    char name[NAME_MAX_SIZE + 1];
    // YOUR LINE HERE
    scanf(format_string, name, &class, &grade);
    return 0;
}
```

What should //YOUR LINE HERE be replaced with?

```c
char *format_string = "Name: %100s Class: %d Grade: %f"
```

Points earned: ______________ out of a possible 2 points
Question 2: Literals ................................................................................. 4 points

(a) (1 point) Which of the following are character literals?

- 4
- 'a'
- '

(b) (1 point) Which of the following are integer literals?

- 4
- 0478
- 0xF3
- '4.5'
- 0
- 7.0

(c) (1 point) Which of the following are floating point literals?

- 4.5
- '4.7'
- 38.7
- 87.0
- 36E4
- 77.0f
- 11.
- 14.81
- four

(d) (1 point) Which of the following are string literals?

- 4.5
- '4.7'
- 38.7
- 87.0
- 36E4
- 77.0f
- 11.
- 14.81
- four

Points earned: __________ out of a possible 4 points
Question 3: Lengths of Arrays .................................................. 4 points

What are the lengths of the arrays for the following array declarations/initializations? If a statement is illegal would be raised, write "illegal".

(a) (1 point) int x[4] = {0, 1, 2, 3, 4} ______ illegal

(b) (1 point) char x[] = "ab\n"11" ______ 7

(c) (1 point) char x[3] = "abcde" ______ illegal

(d) (1 point) float x[5] = {1.3, 4.7, 0.9} ______ 5

Points earned: __________ out of a possible 4 points
Question 4: No Indexing .................................................. 4 points
You need to write a function that returns the number of 'a' characters that appear in a string. Here is its function declaration:

```c
int count_a(char *str);
```

You need to write the function (named "count_a"), but you are not allowed to use the characters [ or ].

**Solution:**

```c
int count_a(char *str) {
    int count = 0;
    for (char * p = str; *p != '\0'; ++p) {
        if (*p == 'a') {
            count += 1;
        }
    }
    return count;
}
```

Points earned: __________ out of a possible 4 points
Question 5: Recursion .................................................. 4 points
Below is a program uses a recursive function. For each of the supplied inputs, write what the program would output. If the program would perform an illegal action, write "illegal".

```c
#include <stdio.h>
int xyz(char *str);

int main(void) {
    char input[10];
    scanf("%s", input);
    int def = xyz(input);
    printf("%d", def);
    return 0;
}

int xyz(char *str) {
    if (*str == '\0') {
        return 0;
    }
    if (*str >= 'a' && *str <= 'z') {
        ++str;
        return 1 + xyz(str);
    }
    ++str;
    return xyz(str);
}
```

(a) (1 point) josh 4

(b) (1 point) RaceTrack 7

(c) (1 point) Joshua_Richard_Nahum illegal

(d) (1 point) My Favorite Class 1

Points earned: __________ out of a possible 4 points
Question 6: String Alteration ................................................. 4 points

I like to make my strings exciting. How do I make a string exciting? I add exclamation points (!) to the end of the string. Sometimes I add a lot of them!!!

I need you to write a function that adds a specified number of exclamation points to the end of a string. You can assume the char array is large enough to hold the additional characters added.

The function takes two arguments, a pointer to a string and an int specifying how many exclamation points to add.

I’ll write the first and last lines for you:

```c
void make_exciting(char * str, int n) {
    // YOUR CODE GOES HERE
}
```

You need to write the rest of the body of the function.

Solution:

```c
// Note: Comments are optional
// Find End Of String
int i = 0;
while(str[i] != '\0') {
    ++i;
}
// i is pointing at the null character
for (int j = 0; j < n; ++j) {
    str[i] = '!' ;
    ++i;
}
// Add the null character to end
str[i] = '\0';
```

Points earned: __________ out of a possible 4 points
If you have finished early, feel free to bring your exam to an instructor.
Or, you can draw a picture of your favorite Pokémon.
Or, you can write a haiku about your love of multidimensional arrays.

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanf Format String</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Literals</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Lengths of Arrays</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>No Indexing</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Recursion</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>String Alteration</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>18</strong></td>
<td></td>
</tr>
</tbody>
</table>