

Name: (3 pts) _____ UMail address (4 pts) _____@umail.ucsb.edu

Lab Section (3 pts) Circle one: 3pm 4pm 5pm

Name of your pair programming partner(s), if you work together:

This assignment is due IN Lab on Thursday 03/04.
It may ONLY be turned in during Lab on Thursday.

Reading: First look at the handout that was given with this homework assignment, or made available to you at the link below:

<http://www.cs.ucsb.edu/~pconrad/cs16/10W/homework/H15/handout> ([printable pdf](#))

This handout goes along with section 6.7 of your Etter textbook, and some exercises that you need to complete online using Ch on CSIL. After you have done the reading and the exercises, answer these questions:

1. This question deals with a session in Ch. Based on what you learned while reading the textbook and the handout, you should be able to predict what Ch would print in each of the cases below where you see *what is printed here?* in light grey italics.

Try to predict the answer in advance and write it down—as if this were an exam question.

Then, before turning in your paper, try this session on Ch to verify whether your answers are correct.

(If you got the answer wrong, change it to the correct answer—but more importantly, try to figure out why you got it wrong—and if you don't understand, come to office hours to ask your instructor or TA for help before the final exam!)

Assume that nothing has been done in the Ch session yet, except perhaps for setting `_prompt = "Ch> "`;

Also, assume that each of the parts of this question (a), (b), (c), etc. occur in a single Ch session—i.e. the variables are remembered from part (a), to (b), etc.

a. (4 pts)

```
Ch> char school[5]="UCSB";
Ch> strlen(school)
what is printed here?
Ch >
```

b. (4 pts)

```
Ch> strlen(school+1)
what is printed here?
Ch>
```

c. (4 pts)

```
Ch> strlen(school)+1
what is printed here?
Ch >
```

d. (4 pts)

```
Ch> school[2] = '\0';
Ch> school
what is printed here?
Ch>
```

- e. (4 pts) Note: remember to consider the effects of what happened in step (d) in the following answers.

```
Ch> strlen(school)
what is printed here?
Ch >
```

f. (5 pts)

```
Ch> school[2] = 'S';
Ch> strlen(school)
what is printed here?
Ch>
```

g. (5 pts)

```
Ch> char team[20];
Ch> strcpy(team,school);
Ch> strlen(team)
what is printed here?
Ch>
```

Please turn over for more problems

Continued from other side...

2. (10 pts) On p. 314 of your textbook, Etter provides two examples of statements for allocating an array called `x`, of size `n`, which can hold `n` double values.

Note that in `n` is a variable (as opposed to a constant or a `#define` of a constant). we can't just do the following:

```
double x[n];
```

C won't let us allocate an array that way unless `n` is a constant. If `n` is a variable, we have to use `malloc` or `calloc`.

However, after the call to `malloc` or `calloc` shown on p. 314, we can use the array `x` *as if* it had been declared as `double x[n];`—that is, even though the variable `x` is declared as a `double *`, once we allocate some memory for `x` to point to, using `malloc` or `calloc`, we can treat it as an array.

Following the example on p. 314, here are two ways to initialize an array of ints called `scores`, with `numScores` values—one solution using `malloc`, and the other using `calloc`. In each case, the right hand side of the `malloc` or `calloc` call is missing. Fill it in.

Solution 1:

```
int numScores
int *scores;

/* ask how many scores */
printf("How many scores? ");
scanf("%i",&numScores);

/* Dynamically allocate memory using malloc */

scores =

if (scores==NULL)
{
    printf("malloc failed\n"); return 1;
}

/* Now we can use scores as an array of int,
   "as if" it were declared this way:  int scores[numScores];  */
```

Solution 2:

```
int numScores
int *scores;

/* ask how many scores */

printf("How many scores? ");
scanf("%i",&numScores);

/* Dynamically allocate memory using calloc */

scores =

if (scores==NULL)
{
    printf("calloc failed\n"); return 1;
}

/* Now we can use scores as an array of int,
   "as if" it were declared this way:  int scores[numScores];  */
```