On the bubble sheet, please write your full name (as it would appear in D2L). You do not need to fill out any of the other fields (ID, section, etc.). Answer the following questions on the bubble sheet:

1. Suppose I declare an array like so:

```
double nums[99];
```

How many slots for doubles does `nums` have?

A. 0
B. 99
C. 100
D. 101
E. we don't know

2. When the following C code is run, what is printed?

```
#include <stdio.h>
void func1(int x) {
    x++;
}
int main(void) {
    int x = 100;
    func1(x);
    printf("x is %d\n", x);
    return(0);
}
```

A. `x is 100`
B. `x is 101`

3. When the following C code is run, what is printed?

```
#include <stdio.h>
void func1(int x[]) {
    x[0]++;
}
int main(void) {
    int x[] = {100}; // this creates an array of size 1 with 100 in the first slot
    func1(x);
    printf("x[0] is %d\n", x[0]);
    return(0);
}
```

A. `x[0] is 100`
B. `x[0] is 101`

4. Suppose a variable `x` is declared like so:

```
int x = 5;
```

What C command would produce a pointer to `x`?

A. `ptr(x)`
B. `*x`
C. `&x`
D. `x*`
E. It's not possible to create a pointer to `x` because it was not declared as a pointer

5. Suppose the file `input.txt` has 5 doubles. If I write my C program `read.c` to read in the doubles using `scanf` and compile my program into an executable called `read`, which of the following would allow me to read in the doubles from `input.txt`?

A. `./read(input.txt)`

B. `./read < input.txt`
C. `./read input.txt`
D. `./input.txt`
E. `./read`

6. Suppose the file `input.txt` has 5 doubles. If I write my C program `read.c` to read in the doubles using `fscanf` and compile my program into an executable called `read`, which of the following would allow me to read in the doubles from `input.txt`?

A. `./read(input.txt)`
B. `./read < input.txt`
C. `./read input.txt`
D. `./input.txt`
E. `./read`

7. Consider the following variable declaration:

```
int *num;
```

What is the data type of `num`?

A. int
B. pointer to an int
C. it hasn't been initialized so it doesn't have a data type yet

8. Suppose the function `add` is defined as:

```
void add(int a, int b, int* result) {
    *result = a + b;
}
```

And suppose we have variables `x`, `y`, and `sum` as follows:

```
int x = 10, y = -2, sum;
```

How can we call the function `add` so that the variable `sum` will hold the value 8 after it is run?

A. `add(*x, *y, *sum)`
B. `result = add(x, y)`
C. `add(x, y, *sum)`
D. `add(x, y, &sum)`
E. `add(&x, &y, &sum)`

9. What does the `sizeof` function do?

A. Gives the size in bytes of a variable
B. Gives the number of slots in an array
C. Gives the amount of data that a pointer points to

10. Is a segmentation fault a compile-time error (syntax error) or a runtime error?

A. Compile-time error
B. Runtime error