

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 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

2. Consider an array of ints declared like so.:

```
int arr[3] = {1, 2, 3};
```

Which of the following could we use to access the third element, 3?

- A. `arr + 2`
- B. `&(arr + 2)`
- C. `*(arr + 2)`
- D. `arr[*arr + 2]`
- E. None of these

3. What is the return type of the function `malloc`?

- A. `int`
- B. `char`
- C. `String`
- D. `pointer`
- E. `double`

4. Suppose I want to allocate space for 10 integers on the heap. Which of the following would do that?

- A. `int intarr[10] = malloc(10 * sizeof(int));`
- B. `int *intarr = calloc(10, sizeof(int));`
- C. `int *intarr = malloc(10);`
- D. `int intarr = (int*) malloc(10 * sizeof(int));`
- E. `int intarr = (int*) malloc(10 * int);`

5. Let the variable `x` be declared and initialized like so:

```
char *x;  
x = calloc(20, sizeof(char));  
strcpy(x, "Hello!");
```

Where does the data stored in `x` live? Hint: the data stored in `x` is a memory address (aka pointer), not the string `Hello`.

- A. On the stack
- B. On the heap
- C. Neither

6. For the same `x` above, where does the H stored at `x[0]` live?

- A. On the stack
- B. On the heap
- C. Neither

7. For the same `x` above, how do I release the memory that it points to back to the operating system?

- A. `delete(*x);`
- B. `delete(x);`
- C. `free(x*);`
- D. `free(x);`
- E. `x.delete();`

8. Consider the following array.

```
double arr[5] = {1.0, 2.0, 3.0, 4.0, 5.0};
```

How can we increase the size of `arr` so that it has space to hold 10 doubles, and copy over the current data that it holds?

- A. `arr = realloc(arr, 10*sizeof(double));`
- B. `arr = malloc(10*sizeof(double));`
- C. `arr = arr[10];`
- D. `int newarr = arr[10];`
- E. We can't

9. Consider the following program:

```
int main(void) {
    int *arr = calloc(10, sizeof(int));
    for (int i = 0; i < 10; i++ ) {
        arr[i] = 10;
    }
    arr[11] = 11;
    free(arr);
    return(0);
}
```

What error does this program have? Select only one.

- A. A syntax error
- B. A memory leak
- C. A memory error
- D. An arithmetic error
- E. It has no errors

10. To check for memory leaks and errors, what program can we use?

- A. `gcc`
- B. `gdb`
- C. `git`
- D. `man`
- E. `valgrind`

11. Suppose that a `planet_t` is a struct with a field called `name`, and that `p` is a pointer to a `planet_t`. What is `p->name` equivalent to?

- A. `p.name`
- B. `strcpy(p.name, x)`
- C. `(*p).name`
- D. `&p.name`