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. What does the linker do in the process of turning C source code into a runnable program?

   A. Allows you to edit C source code.
   B. Runs any preprocesor directives.
   C. Translates C source code into machine code.
   D. Links object files into a final executable.
   E. Runs an executable.

2. When running `gcc`, what does the `-c` flag do?

   A. Produces object files instead of executable files
   B. Runs the linker
   C. Specifies a name for the executable
   D. Compiles C code instead of C++ code

3. Which of the following rules could we put in a Makefile so that `student.o` is recompiled whenever `student.h` or `student.c` is changed?

   A.

```
student.o: student.h student.c
	gcc student.c student.h -Wall
```

   B.

```
student.o:
	gcc student.c student.h -Wall
```

   C.

```
student.o: student.h student.c
	gcc -c student.c -Wall
```

   D.

```
student.o: student.h student.c
	gcc -o student.c main.c -Wall
```

4. Header (`.h`) files are never compiled.

   A. True
   B. False

5. 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

6. When the following code is run, what prints?

```
char str[20] = "Baggins, Bilbo";
strtok(str, ", ");
printf("%s\n", str);
```

A. `Baggins, Bilbo`
B. `Baggins`
C. `Bilbo`
D. `Bilbo Baggins`
E. `Baggins  Bilbo`

7. 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`

8. Suppose that we compile C source code defining main as follows

```
int main(int argc, char* argv[])
```

into an executable called `exe`. If we run `exe` with `./exe 1 2 3`, what is `argv[0]`?

A. `1 2 3`
B. `1`
C. `2`
D. `3`
E. `./exe`

9. Running `exe` as in the previous problem, what is the data type of `argv[2]`?

A. Null pointer
B. String
C. Int
D. Char
E. Double

10. Suppose I run the following:

```
char names[][10] = {"Bob", "Sally", "Mary"};
```

What is `sizeof(names)`? Remember that `char`s take up 1 byte.

A. 3
B. 12
C. 15
D. 18
E. 30

11. Suppose that `int_ptrs` is an array of 10 pointers to ints. For example, it may have been declared using `int* int_ptrs[10];`. How do we get the int that `int_ptrs[0]` points to?

A. `*int_ptrs[0]`
B. `&int_ptrs[0]`
C. `int_ptrs[0]`
D. `int_ptrs[0]*`