**Homework 1 Problem 1**

1. They evaluate to:
   1. 1
   2. 0
2. 15 will be printed to the console.
3. The parentheses () (b) has the highest precedence.
4. Use the variable type `unsigned int`.
5. It will return 9. Comes from 4+3+2 and the fact that you are decrementing the variable x before adding it.
6. “4 2” will be printed without the quotation marks. This comes from the pre-decrement taking effect before the print and the post-decrement happening afterwards.
7. “0 1 2” will be printed to the screen. `continue` completely skips the second print.
8. “<address of x>\_\_9” will be printed. Where <address of x> is determined at runtime.
9. You can access them as follows: `obj.a; pobj->a; robj.a`.
10. Default access permission for structs is `public` and for classes is `private`.
11. No, the main function cannot be overloaded.
12. C) is the correct way to initialize an array of integers.
13. To access the last element use arr[11].
14. A), B), and D) are valid uses of `const`.
15. The following will be printed:

1

3

4

2

Which comes from the order of which the object is created. By definition, class A has to be created before class B. Then the reverse happens for destructors.

1. To free memory use `delete[] a`.
2. Yes, the code is legal as a should be initialized to 0.
3. A) is the only true statement. Pure virtual functions can have a base implementation.
4. Using the keyword `mutable` allows constant class methods to change the value of the variable. In essence, it allows you to whitelist variables than can be altered in a constant instance of your class.
5. The following will be printed:

squash

squash

veggie

squash

This is explained as follows. Both ps and pv are instances of Squash as they use `new Squash()`. The initialization of pv is valid as Squash is a Veggie. Now, the third call returns “veggie” because it explicitly creates a new variable v of type Veggie. Even though you directly initialize it with `v(\*ps)`, it doesn’t matter as the Veggie constructor is being called. Now, the fourth line is a reference assignment. That is, rv is simply a reference to ps which is an instance of Squash. Again, this is valid code because Squash is a Veggie.