CS 180: Data Structures, Spring 2011
Homework 1

Due *via email* by 11:59pm on Monday, February 7

1. (a) Fill in the diagram below to represent the underlying memory configuration that is present after the following commands are executed:

```
int x(34);
int y(x);
int z(22);
int *a(&z);
int &b(x);
```

memory contents	memory address
	262
	263
	264
	265
	266
	267
	268
	269
	270
	271

(b) Now use the diagram below to update the memory configuration from part (a) after the following 5 commands are executed.

```
y = 53;
int *c = new int(43);
a = &y;
x = 17;
*a = 3;
```

memory contents	memory address
	262
	263
	264
	265
	266
	267
	268
	269
	270
	271

2. (Based on Problem C-2.4 from the textbook)

   Write a class Line that implements a line, which is represented by the formula $y = ax + b$. Your class should store a and b as (private) double member variables. In addition, write the following member functions:

   - A constructor that accepts two doubles as input (for a and b). If no inputs are specified, it should default to 0 for both values.
   - The function intersect(ℓ) that takes another line as input and returns the x coordinate at which this line intersects line ℓ. In addition, implement error handling so that if the two lines are parallel, it prints an appropriate error message.
   - A function slope() that returns the slope of the line.

   Finally, write a main function that declares several lines and tests each of your functions.

3. Extra Credit - C-2.3 from the textbook

   Most modern C++ compilers have optimizers that can detect simple cases when it is logically impossible for certain statements in a program to ever be executed. In such cases, the compiler warns the programmer about the useless code. Write a short C++ function that contains code for which it is provably impossible for that code to ever be executed, but your favorite C++ compiler does not detect this fact.