CS 180: Data Structures, Fall 2011
Homework 1

Due *via email* by 11:59pm on Friday, Sept. 14

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

```
int a(52);
int b(22);
int c(a);
int &m(c);
int *x(&a);
```

memory contents	memory address
	281
	282
	283
	284
	285
	286
	287
	288
	289
	290

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

```
int *y = new int(11);
m = 3;
int d = (*x) + 2;
x = y;
a = 6;
```

memory contents	memory address
	281
	282
	283
	284
	285
	286
	287
	288
	289
	290
	291
	292

2. 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 1 for both values.
   - A function slope() that returns the slope of the line.
   - 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 some sort of error checking or handling so that if the two lines are parallel, it prints an appropriate error message (rather than crashing your program).

   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.