

# CS 2150 Exam 1

You **MUST** write your name and e-mail ID on EACH page and bubble in your userid at the bottom of EACH page – including this page.

If you are still writing when “pens down” is called, your exam will be ripped up and not graded – even if you are still writing to fill in the bubble forms. So please do that first. Sorry to have to be strict on this.

Other than bubbling in your userid at the bottom, please do not write in the footer section of each page.

There are 8 pages to this exam – once the exam starts, please make sure you have all 8 pages.

Each page is worth 16 points. Questions are worth 4 or 8 points each, depending on the question length. The first and last pages are worth 2 points each – if you fill in the bubble footer, you get those points. The four point questions on this exam should not take more than a line or two to answer – **your answer should not exceed about 20 words**. There are a total of 100 points of questions on this exam, and 1 hour 45 minutes (105 minutes) to take the exam, which means you should spend about one minute per question point – the remaining 5 minutes are to fill out the bubble footers.

**If you do not bubble in a page, you will not receive credit for that page!**

This exam is CLOSED text book, closed-notes, **closed-calculator**, closed-cell phone, closed-computer, closed-neighbor, etc. Questions are worth different amounts, so be sure to look over all the questions and plan your time accordingly. Please sign the honor pledge below.

---

---

---

---

---

*There are 10 types of people in the world –  
those that understand binary and those that don't.*

---

(the bubble footer is automatically inserted in this space)

**C/C++, page 1**

1. [4 points] Any amount of memory that you allocate via the `new` keyword must be later deallocated via the `delete` keyword. When you declare something without using the `new` keyword, when and how is it deallocated?
  
  
  
  
  
  
  
  
  
  
2. [4 points] Other than syntax, what are the main differences between pointers and references?
  
  
  
  
  
  
  
  
  
  
3. [4 points] Briefly give a convincing example of when we would want to allow implicit constructor initialization in a C++ class.
  
  
  
  
  
  
  
  
  
  
4. [4 points] Briefly, what does the `friend` keyword do in C++? Briefly give a convincing argument (not just an example!) of when we might want to use it.

---

(the bubble footer is automatically inserted in this space)

**C/C++, page 2**

5. [8 points] Consider the code below. The file on the left is odd.h, and the file on the right is even.h. Modify using ONLY preprocessor directives it so that it does not form an #include loop.

```
#include "even.h"  
bool odd (int x);
```

```
#include "odd.h"  
bool even(int x);
```

6. [8 points] Write a SMALL piece of code that shows an example of when you would pass in an `int*` by reference into a subroutine. We aren't looking for a full program here – just a subroutine invocation, the subroutine itself, etc. And it need not be convincing – it just needs to use the fact that the `int*` parameter is passed by reference.

---

(the bubble footer is automatically inserted in this space)



**Numbers**

10. [8 points] Consider the value given, in big Endian hexadecimal format, by 0x40f00000. If this number is an encoded IEEE 754 single-precision floating point number, what is it's decimal representation? Show your work if you want partial credit!

11. [4 points] Briefly describe the general algorithm for how to encode a *negative* number into two's-complement binary notation (assume 32 bits).

12. [4 points] Why did Professor Bloomfield not get any work done this past Sunday evening? It was mentioned in lecture yesterday – were you there? We hope so!

---

(the bubble footer is automatically inserted in this space)

### Arrays

13. [8 points] Consider the declaration `int x[4][2]`. Diagram how this is represented in memory. In particular, it must be clear (via your diagram or an English explanation) where the 2-dimensional cells map to your memory diagram.

14. [4 points] Name two differences between an array base name and a pointer. Specifically, consider the declarations `int x[3]` and `int *y = new int(3)`. Briefly name two differences between how C++ treats the variables `x` and `y`.

15. [4 points] What is the `main()` prototype if you want to access command-line parameters? Briefly, what do each of the two parameters mean?

---

(the bubble footer is automatically inserted in this space)

### Orders of Growth

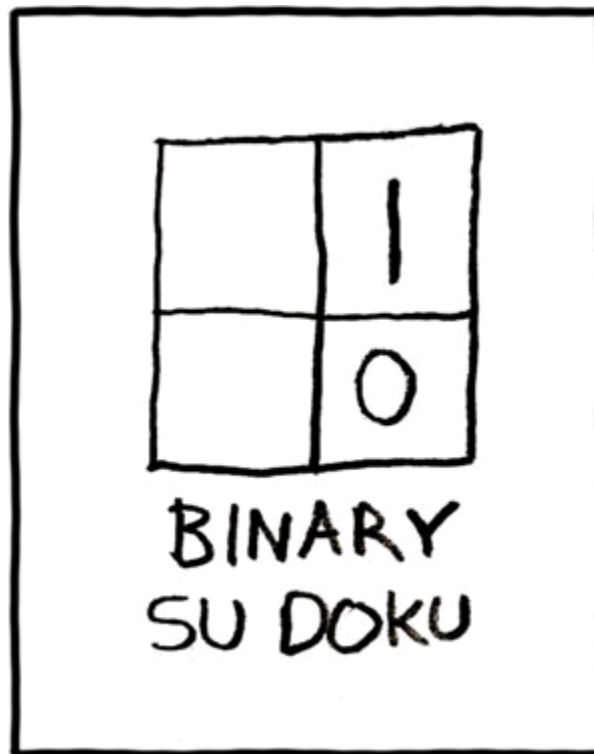
16. [4 points] Briefly, why are the big-Oh and big-Omega analyses not all that particularly useful? Briefly, why do we use big-Theta instead?
17. [4 points] When we write a big-Theta notation, we should not use an equality (i.e.,  $3n = \Theta(n^2)$ ). Briefly, why not? Briefly, what do we use instead? Briefly, why? You can give a single answer that addresses all three of these sub-questions, if that's easier.
18. [8 points] Prove that  $3n$  is  $O(n^2)$

---

(the bubble footer is automatically inserted in this space)

This page unintentionally left blank.

But you get two points for bubbling in the form at the bottom of this page. Woo-hoo!



Can you finish the binary Sudoku? It's not required, but it's quite a challenge!

---

(the bubble footer is automatically inserted in this space)