 Name: _____ Email ID: _____

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 and page 10.

If you do not do this, you will receive a zero for that page! (Or a grade penalty if you leave out the bubble form for pages 1 or 10)

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 10 pages to this exam – once the exam starts, please make sure you have all 10 pages.

There are two types of questions: short answer and long answer. You can tell the difference because the long answer questions have a full page to answer them, while there are 5 short answer questions per page. The 20 short answer questions should not take more than a line or two to answer, and are worth 3 points each. Your answer should not exceed about 20 words. The 4 long answer questions are worth 10 points each. Thus, this entire test is worth 100 points. There is 95 minutes to take it, which means you should spend about 1 minute per question point.

This exam is CLOSED text book, closed-notes, **closed-calculator**, 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 here:

_____

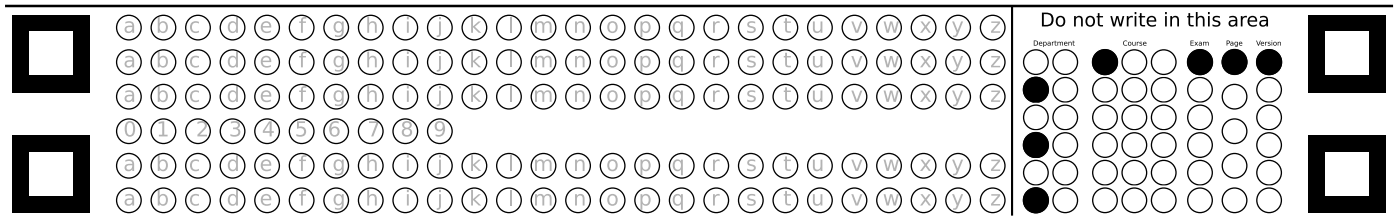
_____

_____

_____

_____

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



 Name: _____ Email ID: _____

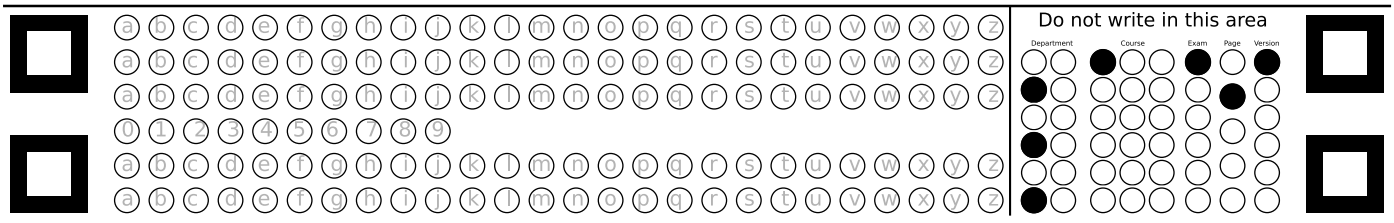
1 What is the Emacs keyboard command to save a file? To exit Emacs?

2 Give two applications of a stack, other than a postfix calculator.

3 Why aren't one's complement binary numbers used often?

4 What is the difference between big-oh and little-oh?

5 What is big-omega? What is big-theta?



 Name: _____ Email ID: _____

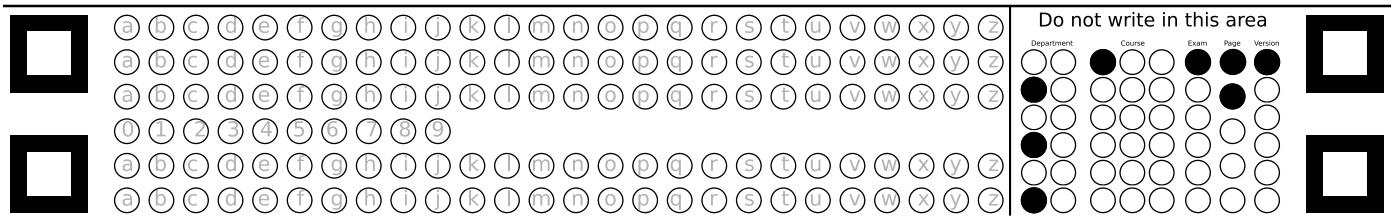
6 There is a little-oh and a little-omega. Why isn't there a little-theta?

7 If an algorithm is O(log *n*), why don't we care about the base of the logarithm?

8 How is a reference different from a pointer?

9 Why do we have the `#ifndef FOO_H / #define FOO_H / ... / #endif` in each header file?

10 What does the `friend` keyword do in C++?



 Name: _____ Email ID: _____

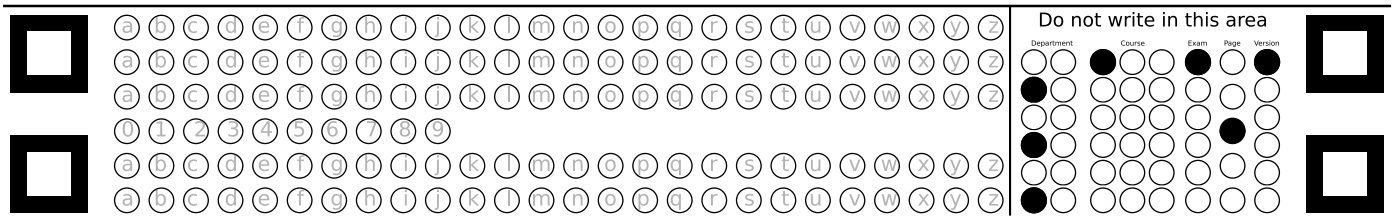
11 What does the `explicit` keyword do in C++?

12 Convert 234 in base 5 to base 9.

13 Convert A4 in base 16 to base 3

14 What do templates allow us to accomplish in C++ that we would not be able to do otherwise?

15 What does a union in C++ accomplish? When would it be used?



 Name: _____ Email ID: _____

16 Fill in the table for the properties of these functions. Put a X in the boxes where the function has the property.

	Reflexive (aRa)	Symmetric (if aRb then bRa)	Transitive (if aRb and bRc then aRc)
Big-Oh			
Big-Omega			
Big-Theta			
Little-Oh			
Little-Theta			

17 Encode -14 as a two's complement binary integer (of 8 bits)

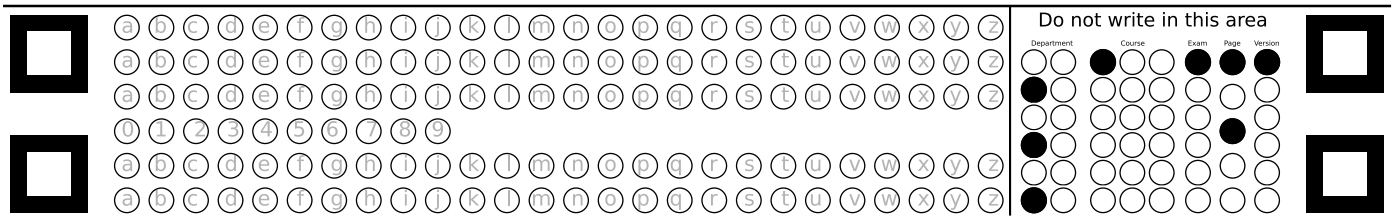
11110010

18 Give a big-Oh estimate for the *recursive* running time of fibonacci(n) (here n means the n^{th} term of the sequence).

19 Give a big-Oh estimate for the *iterative* running time of fibonacci(n) (here n means the nth term of the sequence).

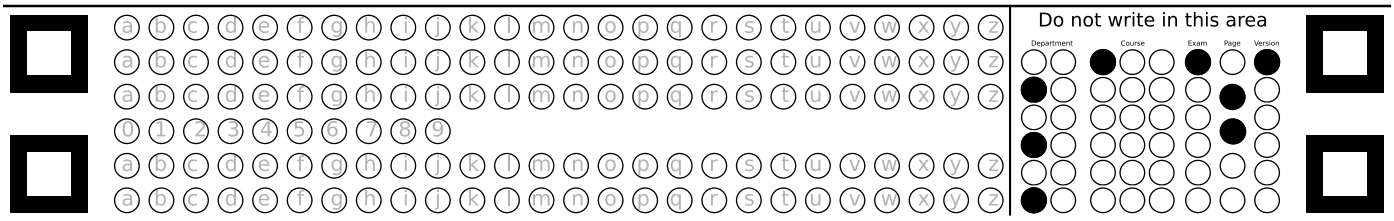
20 What is the exponent and mantissa size for IEEE 754 single-precision and IEEE 754 double-precision (the kind we studied in class) floating point numbers?

single bit 8 bits for exponent and 23 bits for mantissa
double precision: 11 bits for exponent and 52 bits for mantissa

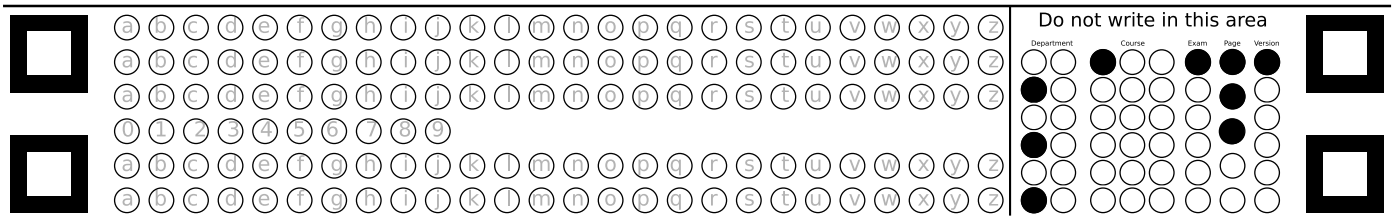


 Name: _____ Email ID: _____

21 Write the queue ADT (i.e. the description of the data structure, as per lecture). State the big-Oh running time for each method.

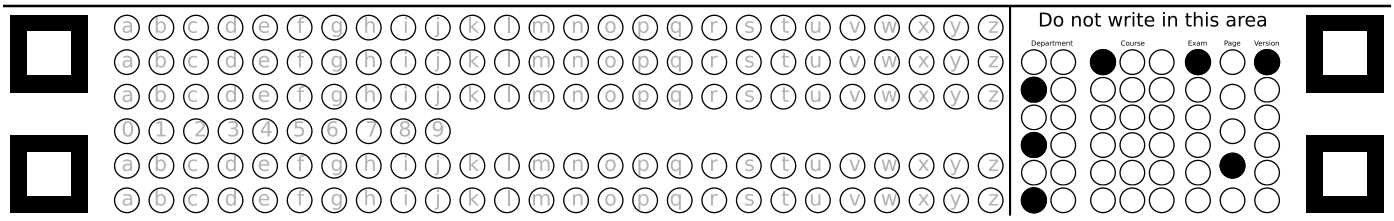


22 Convert -11.5 to the hexadecimal representation for a single-precision IEEE 754 floating point number. You can leave it in big-endian format. You can NOT use a calculator for this. If you don't remember the exponent and mantissa size (i.e. question 20), you can take a guess, and you can still receive partial credit.



 Name: _____ Email ID: _____

23 Write a complete C++ program that will print out all the command-line parameters. As a hint, the prototype for the main method when using command-line parameters is `int main (int argc, char ** argv)`.



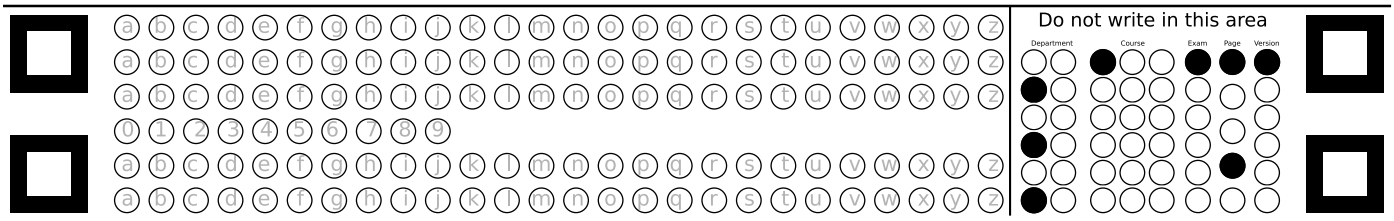
 Name: _____ Email ID: _____

24 What does memory look like after the following code has executed? What is the output? If anything would cause an error, indicate such, and assume that the program would be compiled and executed without that line. Also assume that memory addresses start at 0x1000, and that all ints and pointers are 32 bits. Lastly, prior to running the program, you can assume that all memory has been initialized to a repeating 10101010 pattern. Note that 32 bits of a repeating 10101010 pattern in an int is -1431655766 in base 10.

```
#include <iostream>
using namespace std;

int main() {
  int a = 7;
  int *b;
  int *c = &a;
  int &d = a;
  int &e = *c;
  int **f = &c;
  int g[4];
  int h[] = { 1, 2, 3, 4 };

  cout << a << endl;
  cout << b << endl;
  cout << *b << endl;
  cout << c << endl;
  cout << *c << endl;
  cout << d << endl;
  cout << e << endl;
  cout << f << endl;
  cout << *f << endl;
  cout << **f << endl;
  cout << g << endl;
  cout << g[0] << endl;
  cout << h << endl;
  cout << h[0] << endl;
}
```



 Name: _____ Email ID: _____

This page unintentionally left blank….