NAME: **login id:**

Precept:

COS 126 Midterm 1 Written Exam Fall 2013

This test has 8 questions, weighted as indicated. The exam is closed book, except that you are allowed to use a one page cheatsheet. No calculators or other electronic devices are permitted. Give your answers and show your work in the space provided.

Print your name, login ID, and precept number on this page (now), and **write out and sign the Honor Code pledge** before turning in this paper. *Note*: It is a violation of the Honor Code to discuss this midterm exam question with anyone until after everyone in the class has taken the exam. You have 50 minutes to complete the test.

"I pledge my honor that I have not violated the Honor Code during this examination."

Signature _____

1	/8
2	/10
3	/12
4	/6
5	/8
6	/12
7	/8
8	/6
TOTAL	/70

1. Types and Casts (8 points). Give the type and value of each of the following Java expressions. If an expression will not compile, write `Illegal` under *type* and put an `X` in *value*. You must fill in every entry. Entries left blank will be marked incorrect.

expression	type	value
`( 7 / 2 ) * 2.0`		
`1 + 2 + "0" + 3 * 4`		
`"3" - "2"`		
`true == false`		
`4 < 5 < 6`		
`0.5 + 1/2`		
`Math.max(5.0, 2)`		
`Integer.parseInt("1.5*2")`		

2. Miscellaneous (10 points). Answer the following questions.

A. Which of the following are true of standard input (`StdIn`)? Circle *all* that apply.

(i) No limit on its length.

(ii) Cannot use in any program that uses command-line input.

(iii) Can accept input at any time during program execution.

(iv) Can be intermixed with standard output.

(v) It is an abstraction implemented as a library.

(vi) It is a recursive static method.

B. Which of the following best describes what is a data type? Circle the *single best* answer.

(i) A set of values.

(ii) A set of operations.

(iii) A sequence of 0s and 1s.

(iv) A set of values and operations on those values.

(v) The arguments, name, and return value for a function.

C. What are the reasons that a program might use an array? Circle *all* that apply.

(i) To keep data of different types together in one place.

(ii) To facilitate processing large amounts of data.

(iii) To avoid using `for` loops.

(iv) To represent a vector or a matrix.

(v) To represent a sound wave.

3. Arrays (12 points). Consider the following code.

```
public class ArrayQ
{
   public static void main(String[] args)
   {
      int N = args.length;
      int[] a = new int[N];
         for (int i = 0; i < N; i++)
            a[i] = Integer.parseInt(args[i]);

      for (int i = 0; i < N; i++)
         if (a[i] < a[N-1-i])
         {
            int t = a[i];
            a[i] = a[N-1-i];
            a[N-1-i] = t;
         }

      for (int i = 0; i < N; i++)
         StdOut.print(a[i] + " ");
      StdOut.println();
    }
}
```

For each sequence at left, when `ArrayQ` is run with those numbers on standard input, it will write one of the sequences at right on standard output. Identify which output is produced by each input by writing a letter in each blank. You must fill in all blanks and use all letters (one letter will be used twice).

`3 2 1 4 5` _____

`1 2 3 4 5` _____

`1 3 5 2 4` _____

`1 5 2 4 3` _____

`5 4 3 2 1` _____

`2 3 1 4 5` _____

A. `1 2 3 4 5`

B. `2 3 1 4 5`

C. `3 2 1 4 5`

D. `1 2 5 3 4`

E. `1 4 2 5 3`

4. Redirection and piping (6 points). Consider the following Java program:

```
public class AddX
{
   public static void main(String[] args)
   {
      int N = Integer.parseInt(args[0]);
      int x = Integer.parseInt(args[1]);
      int[] a = new int [N];
      for (int i = 0; i < N; i++)
         a[i] = StdIn.readInt();

      for (int i = 0; i < N; i++)
         a[i] = a[i] + x;

      for (int i = 0; i < N-1; i++)
         StdOut.println(a[i] + " ");

   }
}
```

Assume that the contents of the file `input.txt` are as shown by the following `more` command:

```
% more input.txt
10 20 35 30
```

A. What is the result of the following command? Circle your answer.

```
% java AddX 4 3 < input.txt
```

B. What is the result of the following command? Circle your answer.

```
%  java AddX 3 -1 < input.txt | java AddX 2 3
```

5. Debugging and recursion (8 points). Consider the following program, which is intended to take N from the command line and compute $H_N = 1 + 1/2 + 1/3 + ... + 1/N$.

```
1       public class Harmonic
2       {
3           public static double H(int N)
4           {
5              if (N = 0) return 0;
6              return H(N-1) + 1/N;
7           }
8           public static void main (String[] args)
9           {
10             int N = Integer.parseInt(args[0]);
11             StdOut.println(H[N]);
12          }
13      }
```

In the spaces provided below, identify (*in ten words or less*, each) three bugs and a performance problem in this code. Use the line numbers to refer to the code, and circle your answers. *You do not have to fix the bugs.*

A. A bug that will keep the code from compiling:

B. Another bug that will keep the code from compiling:

C. A bug that will cause the program to print the wrong answer (assuming the compile-time bugs are fixed):

D. A performance problem that makes the program unusable when N is huge (and for which an easy alternative is available):

6. Methods and scope (12 points). Consider the following program, which is not intended to do anything other than test your understanding of functions, scope and side effects.

```
public class FunctionQ
{
   public static int[] f(int[] x, int a)
   {
      for (int i = 0; i < x.length; i++)
         x[i] *= a;
      a = x[2];
      return x;
   }

   public static void main(String[] args)
   {
      int[] x = {1, 2, 3};
      int a = 4;
      int[] y = f(x, a);
      y[0] = 0;
      StdOut.println(a);
      StdOut.println(x[0]);
      StdOut.println(y[0]);
      StdOut.println(x[1]);
   }
}
```

Give the output of this program in the blanks provided below.

A. First line: _____

B. Second line: _____

C. Third line: _____

D. Fourth line: _____

7. **Performance (8 points).** Each lettered item below specifies a quantity that is a function of N. Identify the order of growth (with respect to N) of each one by writing one of the words *linear, quadratic, cubic* or *exponential* in the corresponding blanks at right.

A. Memory used by the code

```
boolean[] myArray = new boolean[N*N*N];
```

_____

B. Time required to execute the code

```
int[] myIntArray = new int[N];
```

_____

C. Number of lines printed by the code

```
for (int i=0; i<N; i++)
   for (int j=0; j<3; j++)
      StdOut.println("COS 126");
```

_____

D. Number of lines printed by the code

```
for (int i=0; i<N; i++)
   for (int j=N; j>0; j--)
      StdOut.println("COS 126");
```

_____

E. The time it takes to draw a Sierpinski triangle of order N

_____

F. The order of growth of the running time of a program that runs for 3 days when N is 100000, 24 days when N is 200000, and 6.5 months when N is 400000.

_____

G. The order of growth of the running time of a program that runs for 4 seconds when N is 100,000, 10 seconds when N is 250,000, and 1 minute when N is 1,500,000.

_____

H. Memory used by the code

```
String x = "hi";
for (int i=0; i<N; i++) x += x;
```

_____

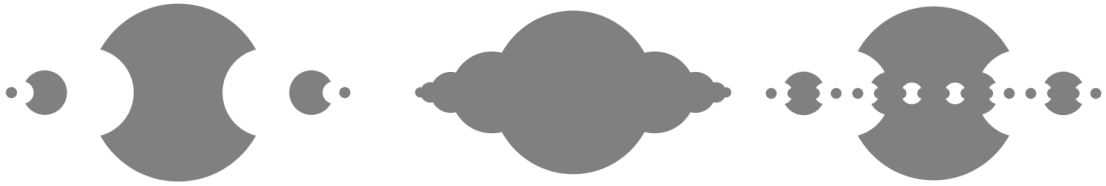
8. Recursion (6 points). Consider the following program, which includes a recursive method:

```
1  public class Pattern
2  {
3     public static void draw (int n, double r, double x, double y)
4     {
5        if (n == 0) return;
6
7        if (n % 2 != 0) StdDraw.setPenColor(StdDraw.GRAY);
8        else StdDraw.setPenColor(StdDraw.WHITE);
9
10       draw(n-1, r/2, x-r, y);
11
12       draw(n-1, r/2, x+r, y);
13
14    }
15
16    public static void main (String[] args)
17    {
18       draw(5, .25, .5, .5);
19    }
20 }
```

Suppose we add the line

```
StdDraw.filledCircle(x, y, r);
```

to the program either on line 9, 11, or 13 (before, between, and after the recursive calls respectively). If we do so, each version produces one of the diagrams below.



Circle the correct answer (left, center or right) in each row. Some answers may be used more than once.

If we add the command on line 9, which diagram is generated? left center right

If we add the command on line 11, which diagram is generated? left center right

If we add the command on line 13, which diagram is generated? left center right